BOOK VII

Minivac 601/6010

Capacitor/Resistor Manual

Copyright © 1962
by
SCIENTIFIC
DEVELOPMENT
CORPORATION
Watertown
Mass.

. . . Creative circuits, expanded arithmetic capabilities and more sophisticated programs using capacitors and resistors

MINIVAC 601/6010

Capacitor/Resistor Manual

EDITED BY LINDA G. SPRAGUE

SCIENTIFIC DEVELOPMENT CORP.
WATERTOWN, MASS.

First Printing—January, 1962

Copyright © 1962 by Scientific Development Corporation, 372 Main Street, Watertown, Massachusetts

CONTENTS

	PAGE NUMBER
Preface	
Installation	1
Capacitors	1
Resistors	2
What is Electrical Resistance?	2
Ohm's Law	4
Resistor Notation	4
The Resistor Color Code	5
The Use of Resistors	6
More Than One Resistor	8
Selecting a Bias Resistor for Each MINIVAC Relay	8
Capacitance and Capacitors	9
Capacitor Notation	11
The Use of Capacitors	11
More Than One Capacitor	11
Notes on the Use of Capacitors and Resistors in MINIVAC Circuits	12
Experiment 1: Simple Flip-Flop	12
Experiment 2: Fast Flip-Flop	13
Experiment 3: Oscillator	14
Experiment 4: Delay Flop — Pulse Input	15
Experiment 5: Delay Flop — Continuous Input	17
Experiment 6: Six-Bit Binary Counter	19
Experiment 7: Automatic Five-Bit Binary Counter	20
Experiment 8: Counter for Bases 9, 10, 11, 12, 13, 14, 15 and 16	21
Experiment 9: Decimal to Binary Converter	24
Experiment 10: Multiplication with a Counter	25
Experiment 11: Division with a Counter	26
Experiment 12: Combined Arithmetic Program	27
Experiment 13: Four-Bit Automatic Shift Register	29
Experiment 14: Multiplication with an Accumulator	31
Experiment 15: Division with an Accumulator	32
Experiment 16: Ring-Tail Switch-Tail Shift Register	36
Experiment 17: Electronic Memory	37
Experiment 18: ESP Tester	38
Experiment 19: Tic-Tac-Toe	40
Experiment 20: Music Composer I	41
Experiment 21: Music Composer II	44
Experiment 22: Team Reaction Tester	45
Experiment 23: Black Jack	46
Experiment 24: Script Writer for Westerns	47

BOOK VII
Capacitor/Resistor Manual

PREFACE

The addition of capacitors and resistors to the basic MINIVAC extends the computer's capabilities to include more of the circuits used in large-scale digital computers, as well as expansions of those circuits which can be built on the MINIVAC alone.

This manual assumes a familiarity with the basic MINIVAC components and their use such as would be gained through the use of the MINIVAC Manual, Books I through VI.

NOTE: If you have a MINIVAC 6010, the "Installation" section of this manual does not apply; capacitors are already installed on the MINIVAC 6010.

INSTALLATION

Capacitors

The six capacitors must be permanently mounted beneath the MINIVAC console.

To install the capacitors, remove the main console panel:

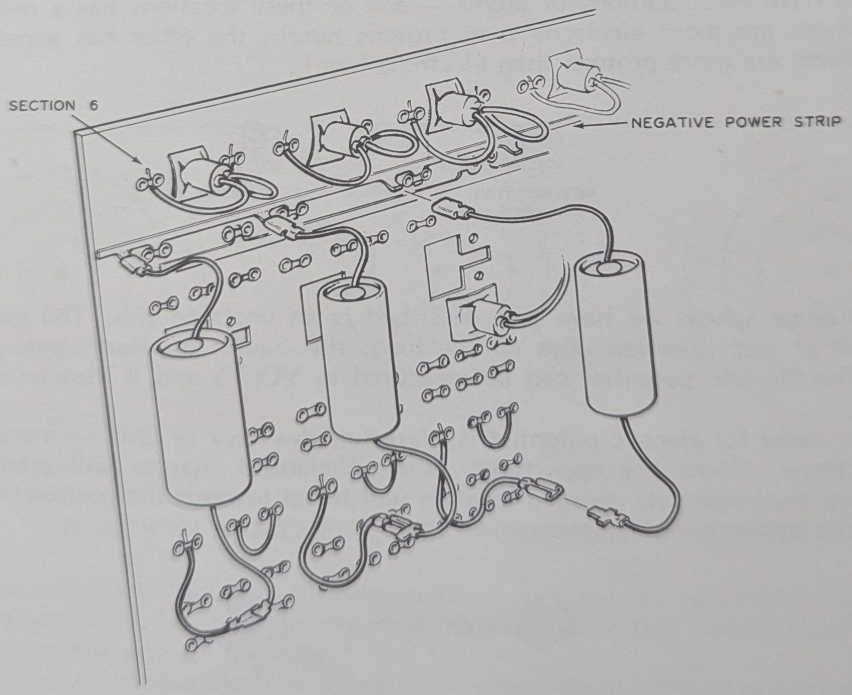
1. Be sure that the MINIVAC is NOT plugged in.
2. * Remove the front panel of the unit.
3. Disconnect the power panel from the main panel by unhooking the connections from the power supply to the power strips.
4. Lift the main panel out of the computer case.

*To reach the underside of wooden-cased MINIVACs, remove the bottom panel by unscrewing the rubber feet.

Notice that along the metal strip connecting the negative (−) power terminals there are six flat tabs. Bend these tabs up so that they will accept the connectors on the capacitors.

One of the capacitors is larger than the other five. This capacitor is to be installed under Section 6 (the section nearest the power panel and the rotary switch mechanism). Mount this larger capacitor first by sliding the matching end of the capacitor's connecting wire onto the negative power tab under Section 6. With this connection in place, slide the tab on the free end of the capacitor into the connecting wire from the 6 cap terminal. Be sure that the connections are tight.

The other five capacitors are mounted in the same way:



VIEW OF UNDERSIDE OF COMPUTER PANEL SHOWING INSTALLATION OF CAPACITORS

The weight of the capacitors causes strain on the connections. For more reliable operation, loop the capacitor's connecting wire around the relay wires; then cement the capacitors to the underside of the panel. Slide the main panel into the computer case and replace the power connections. Replace the front panel.

Resistors

The resistors are ready for use on the computer console. The resistors, with programming wires attached, are used in place of ordinary programming wires for certain program connections. Their use is explained in the following section.

WHAT IS ELECTRICAL RESISTANCE?

Electrical resistance is defined as a ratio between ***voltage*** and ***current.*** Its actual effect can be described in terms of the atomic particles, electrons and protons.

Complete atoms have an equal number of protons and electrons. Since electrons carry a negative charge and protons carry a positive charge, an atom with equal numbers of each has no net charge. For example, the net charge on an oxygen atom which has the atomic number 8 (which means that the atom contains 8 protons) is:

-8 (negative electrons) plus $+8$ (positive protons)

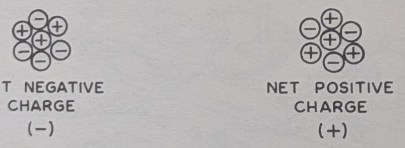
or

$-8 + 8 = 0$ net charge

(Charge is measured in ***coulombs;*** the charge on an electron is 1.6×10^{-19} coulombs or .00000000000000000016 coulombs.)

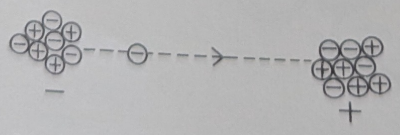
Now consider the following situation:

we have two locations of atoms — one of these locations has a net negative charge (there are more electrons than protons here); the other has a net positive charge (there are more protons than electrons here):

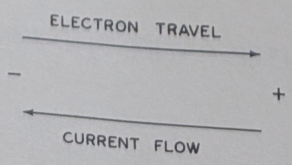


The situation which we have just described is an unstable one. The number of protons and electrons at each location does not balance; this causes an ***electric potential*** between the locations. The electric potential can be measured in VOLTS and is also known as a ***voltage difference.***

Another name for electric potential is ***electro-motive force*** or EMF — literally, the force for moving electrons. Given the opportunity, the unbalanced charges will attempt to stabilize; electrons from the negatively charged location will travel towards the positively charged location because of the attraction of opposites:



The rate of flow of electrons in such a situation is called ***current.*** Although the electrons travel from − to +, current is generally shown from + to −:

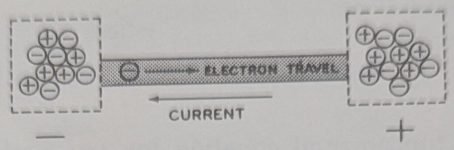


(The notation for current was originally used by Benjamin Franklin. He correctly assumed that one of the charged particles moved towards the particles of opposite charge; however, he incorrectly assumed that the positive charge moved towards the negative charge.)

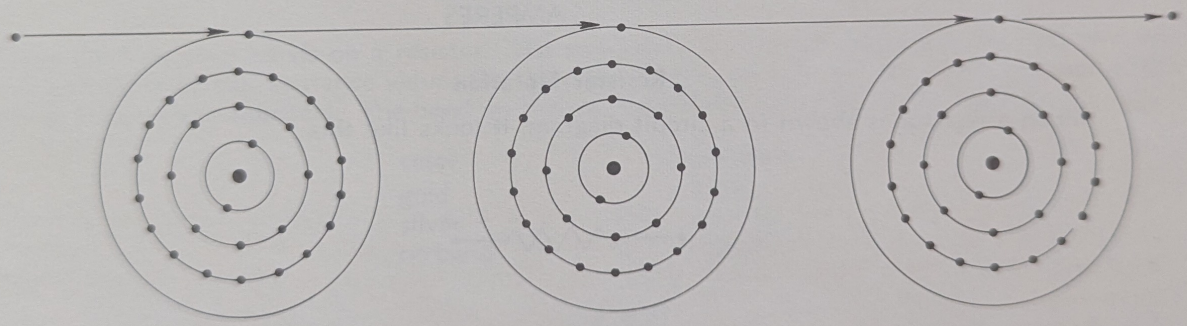
Current — the rate of flow of electrons — is measured in AMPERES or AMPS:

$$\text{AMPS} = \frac{\text{COULOMBS}}{\text{SECOND}}$$

However, electrons must have a medium through which to travel. We could, for instance, place a tube of gas, liquid or solid between the locations of opposite charge. The electrons would then flow through this tube towards the positive charge:



If the tube is already filled with atoms — as in the sketch above — the electrons will find obstacles in their path of travel. The atoms in the tube, although necessary for the electrons' travel, hinder their flow. To overcome this difficulty, the electrons bump into outermost electrons of the atoms, replace them and send them traveling towards the positive charge:



ELECTRON MOVEMENT THROUGH ATOMS OF COPPER

This is not unlike the "billiard ball technique" where a moving ball collides with a stationary ball, stops and sets the other ball moving. Electrons continue this "billiard ball" pattern until an electron leaves the end of the tube.

Some substances oppose electron travel more than others. The quantitative measurement of this is called ***electrical resistance***.

Ohm's Law

Resistance is defined as a ratio between voltage and current for a given conducting body. If various amounts of current are sent through a conducting body — for instance, a piece of wire — the voltage difference between the ends of the wire will be proportional to the current according to the equation:

$$\frac{V}{i} = \text{constant}$$

where: V = voltage in volts
i = current in amps

This constant can be measured for a given conducting body either by sending a known amount of current through it and measuring the voltage difference across it, or by applying a known voltage difference across it and measuring the current flow.

The constant ratio is called the ***resistance*** of the conducting body, and is represented by the letter R. So the equation we have just stated becomes:

$$\frac{V}{i} = R$$

where: V = voltage
i = current
R = resistance

This is OHM's LAW — a fundamental equation for any work involving electric current flow (electricity). Ohm's Law can be expressed in three forms:

$$V = iR \qquad R = \frac{V}{i} \qquad i = \frac{V}{R}$$

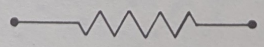
so that if any two of the three factors are known, the third can easily be calculated.

Resistance is measured in OHMS:

$$\text{OHM} = \frac{\text{VOLTS}}{\text{AMPERES}}$$

Resistor Notation

When a resistor is shown in a circuit diagram, it looks like this:



The value of a resistor is given in OHMs. The greek capital letter omega (Ω) is often used to represent "ohms"; we will use it to represent both a resistor and its value according to the following convention:

a) the symbol "Ω" will be used to represent a resistor
b) the subscript to the Ω will indicate the value of the resistor

For example, Ω_{100} means a 100 ohm resistor; Ω_{150} means a 150 ohm resistor. (Note: a special case — the bias resistor for a particular relay — will be designated as Ω_{B1}, Ω_{B2}, etc.)

The resistors provided for the MINIVAC have programming wires attached and are to be used ***in place of*** regular programming wires for certain connections. To indicate the use of a resistor in a MINIVAC program, we will use two consecutive program instructions enclosed by parentheses. For example:

$$\begin{pmatrix} 1F/\Omega_{150} \\ \Omega_{150}/1J \end{pmatrix}$$

means: use a 150 ohm resistor to make the connection between 1F and 1J. Similarly:

$$\begin{pmatrix} 1X/\Omega_{22} \\ \Omega_{22}/1H \end{pmatrix}$$

means: use a 22 ohm resistor to make the connection between 1X and 1H.

The Resistor Color Code

The resistance value of commercial resistors is often printed on the component. Some commercial resistors, however, are marked with a color code to show the value of the resistance in ohms. Colored bands are used to represent digits according to the convention:

color	value
black	0
brown	1
red	2
orange	3
yellow	4
green	5
blue	6
violet	7
grey	8
white	9

There are four color bands on a resistor. The two bands on the left represent the first two significant digits of the resistance value; the third band from the left represents the number of 0's which follow these numbers. The band on the right gives the tolerance of the resistance:

color	tolerance
gold	±5%
silver	±10%
no band	±20%

For example:

RED-VIOLET	represents 27
BROWN	represents one "0"
SILVER	represents ±10%

This resistor has a resistance of:

270 ohms ±10%

Or:

RED-RED	represents 22
BLACK	represents no "0"
GOLD	represents ±5%

This resistor has a resistance of:

22 ohms ±5%

The Use of Resistors

In general, resistors are used when it is necessary or desirable to control a voltage-current ratio. For example, if the voltage between two points is fixed, the current flowing from one point to the other depends upon the resistance between the two points. The current may be varied by changing the resistance of this current path. As an example of why it would be desirable to vary current flow, we will examine the MINIVAC's relays.

We have seen that the relay with its associated contacts is a bi-stable element; that is, it can be wired to stay in either of two positions — on or off. We turn a relay ON by supplying sufficient current to it, or we could say, by providing sufficient voltage across the relay coil. Conversely, we turn a relay OFF by decreasing the current to it — that is, by removing the voltage difference across the relay coil — until the relay turns off.

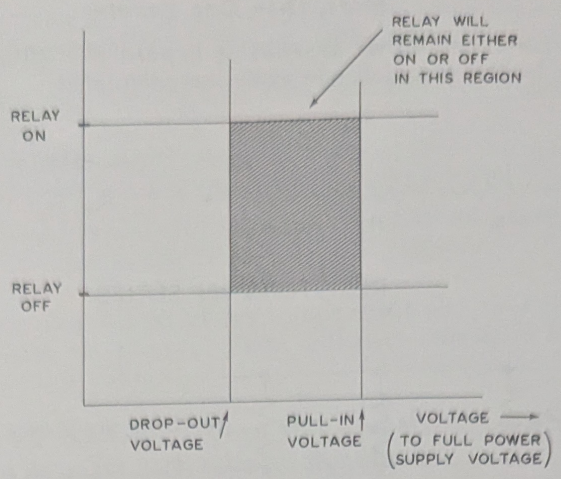
However, in order to function as a bi-stable element, the relay must be able to ***remain*** either on or off once it is positioned. This requires the use of a normally-open contact — either G and H or K and L — which leaves only one contact free for additional circuitry. To increase the capabilities of the relay, it would be desirable to have both contacts (G, H and J and K, L, and N) available. This can be accomplished through the appropriate use of a resistor.

Consider what happens when a relay coil is energized; current flows through the coil, producing a magnetic force which pulls the relay arm against the core of the coil. The switching motion of the arm is basically dependent upon the amount of magnetic force produced in the relay coil; this in turn is dependent upon the current to the relay coil and the gap between the core of the coil and the relay arm. Specifically, the greater the magnetic force the more likely that the arm will pull in.

The current in a relay coil is directly proportional to the voltage across the coil ($V = \frac{i}{R}$). We can therefore say that the magnetic force in a relay coil is dependent upon the voltage across the coil and the gap between the core and the arm. For a given gap, the greater the voltage the greater the magnetic force; for a given voltage, the smaller the gap, the greater the magnetic force.

Since the gap on a closed relay is very small, less voltage is required to hold it in than is required to pull it in. And, the voltage at which a relay turns off is less than the voltage at which it turns on. The voltage at which a relay turns on is called the "pull-in" voltage; the voltage at which a relay turns off is called the "drop-out" voltage. The MINIVAC's relays are deliberately designed so that there is a great difference between the "pull-in" and "drop-out" voltages.

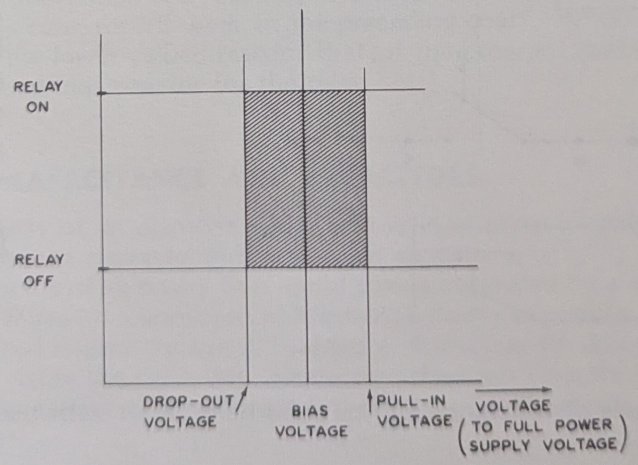
If we gradually increase the voltage across a relay coil, we will eventually reach a "pull-in" voltage for the relay. If we now gradually decrease the voltage across the relay coil, the relay arm will "drop-out" at a lower voltage. We can show this graphically as:



From this sketch we can see that there is region between the "drop-out" and "pull-in" voltages where the relay may be ***either*** on or off — the shaded area in the graph. In this region the relay is "bi-stable" — it will remain either on or off. If we could make use of this region, we could use the relay as a bi-stable element, yet leave its relay contacts free for additional circuitry.

To realize this advantage, we can use a resistor to determine the amount of voltage which is supplied to the relay coil. By placing an appropriate resistor between power and the relay coil, we can insure that the voltage supplied to the relay will fall within the region defined by the "pull-in" and "drop-out" voltages. Then, if we constantly supply power to the relay coil, the relay will be a bi-stable element and both sets of contacts will be free.

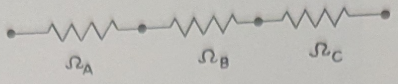
The mid-region voltage is the "bias" voltage for the relay. The voltage to the relay can be limited to the bias voltage by using a "bias" resistor:



With power supplied to the relay coil through the bias resistor, the relay will remain either on or off once it is in position. We will use a capacitor to position the relay — that is, to turn it on or off.

More Than One Resistor

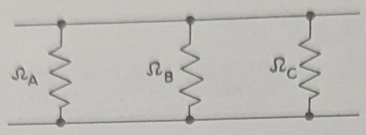
Resistors may be combined either ***in series*** or ***in parallel.*** When resistors are combined, the effective resistance of the combination can easily be calculated:



TOTAL RESISTANCE = Ω_T

$$\Omega_T = \Omega_A + \Omega_B + \Omega_C$$

RESISTORS IN SERIES



TOTAL RESISTANCE = Ω_T

$$\frac{1}{\Omega_T} = \frac{1}{\Omega_A} + \frac{1}{\Omega_B} + \frac{1}{\Omega_C}$$

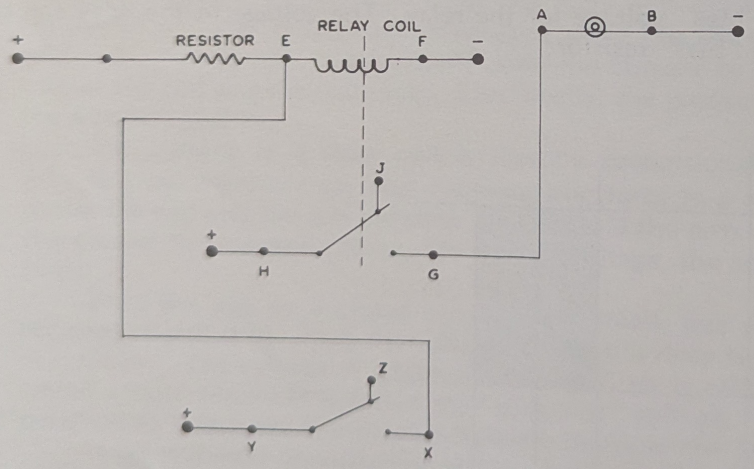
RESISTORS IN PARALLEL

SELECTING A BIAS RESISTOR FOR EACH MINIVAC RELAY

To use the MINIVAC relays as bi-stable elements without using the relay contacts, we must select the bias resistor for each relay.

THE FOLLOWING INSTRUCTIONS ARE FOR A SINGLE RELAY. THIS PROCEDURE MUST BE CARRIED OUT FOR EACH INDIVIDUAL RELAY.

The program and circuit diagram for bias resistor selection are:



A/G
B/–
E/X
F/–
H/+
Y/+
$\begin{pmatrix} +/\Omega \\ \Omega/E \end{pmatrix}$*

* Decreasing values of resistance will be tried here. Start with the highest valued resistor in the kit.

Now make a chart like this:

RESISTANCE	RELAY OFF	RELAY ON
Ω =		
Ω =		
Ω =		

Enter at the top of the "RESISTANCE" column the value of the resistor which you have wired into the program. (Start with the highest possible value.) Now follow these steps:

1. Turn power on. If the relay remains OFF (the light will remain off), place a check in the "RELAY OFF" column.
2. Push and release the pushbutton. If the relay remains ON (the light will remain on) after the pushbutton is released, place a check in the "RELAY ON" column.
3. Remove the resistor which is wired into the circuit. Replace it with the next highest value resistor and enter this value in the "RESISTANCE" column.

Repeat these steps using decreasing values of resistance until you find the lowest resistance at which the relay will remain ***both*** off and on. (You will have checks in both the "RELAY OFF" and "RELAY ON" column for this value.)

This is the ***bias resistor*** for the relay. When you have carried out this procedure for each relay, enter the bias resistance value for each relay on this chart:

relay	value of bias resistance (in ohms)
1	Ω_{B1} =
2	Ω_{B2} =
3	Ω_{B3} =
4	Ω_{B4} =
5	Ω_{B5} =
6	Ω_{B6} =

(Notice that the notation for a particular relay's bias resistance is given as "omega" with the subscripts B1, B2, etc. When given as a program instruction, two consecutive connections will be stated:

$$\begin{pmatrix} 1F/\Omega_{B1} \\ \Omega_{B1}/1J \end{pmatrix}$$

This means: use the bias resistance for relay 1 to connect 1F and 1J.

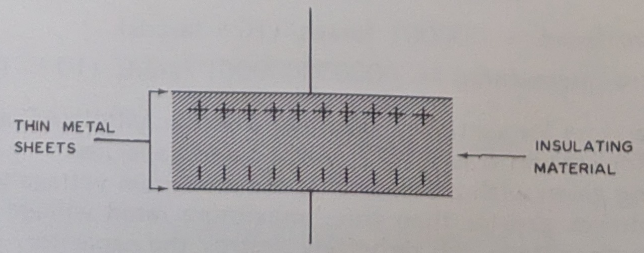
Note: You may find that more than one value of resistance will meet the requirements of the bias resistor for a relay. In this case, record ***both*** in the preceding chart. When using that relay's bias resistor in a program, try the lower valued resistor first; if the program does not operate properly, substitute the higher valued bias resistor for the relay.

CAPACITANCE AND CAPACITORS

Capacitance is that property of an element which allows it to store electrical charge. Components which exhibit this storage characteristic are called capacitors.

A capacitor generally consists of two very thin metal sheets separated by a sheet of insulating (non-conducting) material. Wires are connected to the metal sheets to provide the terminals for the capacitor. (The sandwiched sheets are usually folded or rolled up to save space.)

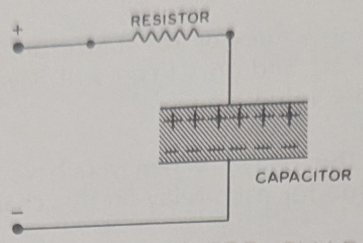
If direct current is supplied to the capacitor, the current does not flow through it. However, equal amounts of positive and negative charge accumulate on the metallic sheets. Then, a charged capacitor may be presented as:



We can see that at this point one of the "plates" (metallic sheets) is positively charged and the other plate is negatively charged. That is, one side of the capacitor is more negative than the other (or, we would say that one side is more positive than the other). This means that there is a ***voltage difference*** between the plates of a charged capacitor.

If we now remove the direct current source, the plates of the charged capacitor will hold this charge. In effect then, we have a miniature battery which is storing a potential voltage. To use this voltage, we simply provide a path of travel for the current which it will produce by connecting the ends of the capacitor together (sometimes through another component).

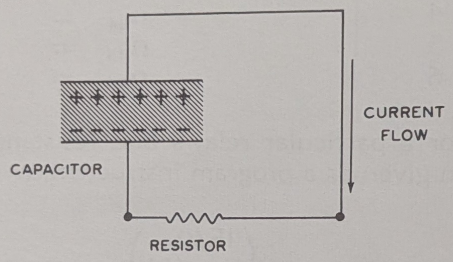
A generalized circuit diagram for charging the plates of a capacitor looks like this:



GENERALIZED CIRCUIT DIAGRAM FOR CHARGING A CAPACITOR

(A resistor is frequently used to limit the flow of current to a capacitor.)

Similarly, a generalized circuit diagram for discharging a capacitor (using the voltage stored in a charged capacitor) is:



GENERALIZED CIRCUIT DIAGRAM FOR DISCHARGING A CAPACITOR

(A resistor is frequently used to limit the flow of current out of a capacitor.)

The ***capacity*** of a capacitor is the amount of charge which it will hold for a given voltage. Specifically,

$$C = \frac{Q}{V}$$

where C = capacity
Q = charge
V = voltage

This means that capacity is measured in "coulombs per volt". For convenience, we call this unit a FARAD. That is,

$$\text{FARAD} = \frac{\text{COULOMBS}}{\text{VOLTS}}$$

Commercial capacitors are generally available in "micro-farad" or "micro-micro-farad" units.

1 micro-farad = .000001 farads (10^{-6} farads)

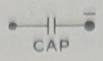
1 micro-micro-farad = .000000000001 farads (10^{-12} farads)

For example, the capacitors for sections 1 through 5 of the MINIVAC are 500 micro-farad capacitors. The capacitor for section 6 is a 1000 micro-farad capacitor.

The voltage rating given with a capacitor is the ***maximum*** voltage which can safely be applied to the capacitor. Voltages greater than this "maximum rated voltage" are liable to destroy the capacitor; a high enough voltage will definitely destroy the capacitor.

Capacitor Notation

We will use this symbol to represent a capacitor in a MINIVAC circuit:



Note: Since one end of the capacitor is permanently connected to the negative power source, we will always show negative power on one end of the capacitor. No connection need be made between the capacitor and negative power.

As indicated in Book 1 (page 25) there are terminals on the MINIVAC console for the capacitors. These terminals will be indicated in a program by:

1cap, 2cap, 3cap, etc.

Connections to the capacitors are usually made with the regular programming wires, although when indicated a resistor may also be used. For example,

$$\begin{pmatrix} 1+/\Omega_{220} \\ \Omega_{220}\,1/\text{cap} \end{pmatrix}$$

means: use a 220 ohm resistor to make the connection between 1 cap and 1+.

The Use of Capacitors

Because capacitors can store and release electric energy, they can be used to turn relays on and off. This is their principal application in MINIVAC circuits — providing the voltage changes required to pull in or drop out a biased relay.

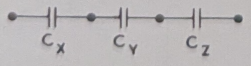
A ***charged*** capacitor is connected across the relay coil to pull in a biased relay. The brief surge of power from the capacitor increases the voltage across the relay coil, causing the relay arm to pull in. The relay remains on after the capacitor's energy has been discharged because of the bias voltage supplied to it.

A relay can be dropped out by connecting a discharged capacitor across the energized relay coil. As energy flows from the relay coil into the capacitor, the capacitor becomes charged and the decrease in energy in the relay coil causes the relay arm to drop out. Since the relay is biased, the normal voltage across it is less than the pull-in voltage; therefore, the relay will remain off.

In general, capacitors are used whenever a brief surge of voltage is required; or, when it is necessary to briefly decrease a voltage.

More Than One Capacitor

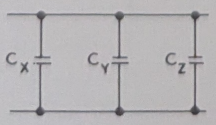
Capacitors can be combined either ***in series*** or ***in parallel.*** The effective capacitance of a combination of capacitors can easily be calculated:



TOTAL CAPACITANCE = C_T

$$\frac{1}{C_T} = \frac{1}{C_X} + \frac{1}{C_Y} + \frac{1}{C_Z}$$

CAPACITORS IN SERIES



TOTAL CAPACITANCE = C_T

$$C_T = C_X + C_Y + C_Z$$

CAPACITORS IN PARALLEL

NOTES ON THE USE OF CAPACITORS AND RESISTORS IN MINIVAC CIRCUITS

A resistor (33 ohms or less) may be connected between D18 and D19 to make the rotary switch mechanism operate more slowly.

When a capacitor is connected across a relay coil, the "pull-in" and "drop-out" characteristics of the relay coil may change. This may require the use of a different bias resistance.

A capacitor should not be connected to positive (+) power. The resulting increase in voltage may burn out the lights.

The 1000 micro-farad capacitor under section 6 of the MINIVAC is used in the construction of a slow-speed oscillator. To build a high-speed flip-flop on section 6, use a 500 micro-farad capacitor from sections 1 through 5.

Do not touch resistors which are wired into a program when the power is on. When current flows through a resistor, the resistance causes the elements to heat up. Resistors in use are hot.

Experiment 1: Simple Flip-Flop

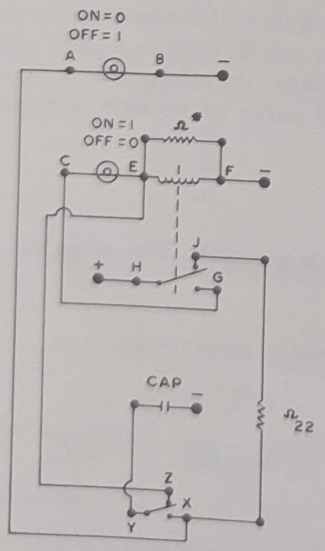
THIS FLIP-FLOP CAN BE BUILT ON ANY OF THE SIX SECTIONS OF THE MINIVAC.

When a capacitor is used, a flip-flop can be constructed using only one relay. (For other flip-flop circuits see: Book II, page 11 ff.; Book IV, page 68 ff.) An advantage of this single-relay flip-flop is that it requires the use of only one set of relay contacts, leaving the second set of contacts free for additional circuitry (see circuit diagram).

In this circuit the capacitor discharges through a binary output light; the light is used to provide a high resistance. Since the capacitor must completely discharge through a high resistance, this flip-flop cannot be operated at very high speeds. (Notice that neither of the lights in this circuit operates at full brightness; this is because both are connected with resistances in series which reduces the amount of current flowing through them.)

Two lights — a binary output light and a relay indicator light — are used to indicate the state of the flip-flop. These lights are ***complements;*** that is:

Binary Output Light	ON = 0	OFF = 1
Relay Indicator Light	ON = 1	OFF = 0



A/X
B/−
C/G
E/Z
(E/Ω
Ω/F)*

F/−
H/+
(J/Ω_{22}
Ω_{22}/X)
Y/cap

*Note: this is an ***optional*** connection. If the flip-flop does not operate reliably, make this connection: start with a 150 ohm resistor and use increasingly higher values of resistance until the flip-flop operates to your satisfaction.

Procedure:

1. Turn power on. The binary output light comes ON; the relay indicator light remains OFF. The flip-flop is now storing "zero" (0).
2. Push the pushbutton and hold it down. The capacitor is now charging through the 22 ohm resistor.
3. Release the pushbutton. The relay indicator light comes ON; the binary output light goes OFF — the flip-flop is now storing "one" (1). The energy which accumulated in the capacitor while the pushbutton was down has been transferred to the relay coil. After the energy from the capacitor has pulled the relay in, the relay is held on by the power supplied through the normally open contact.
4. Push the pushbutton and hold it down. The capacitor is now discharging through the resistance provided by the binary output light. Hold the pushbutton down long enough to allow the capacitor to completely discharge.
5. Release the pushbutton; the flip-flop is now storing "zero" (0). The energy in the relay coil was directed to the capacitor and the relay arm dropped out. However, the relay dropped out before the capacitor could fully charge.
6. Try operating this flip-flop at increasing speeds. Notice that if the flip-flop is operated too fast to allow the capacitor to fully charge or discharge, the flip-flop will begin to miss.

Experiment 2: Fast Flip-Flop

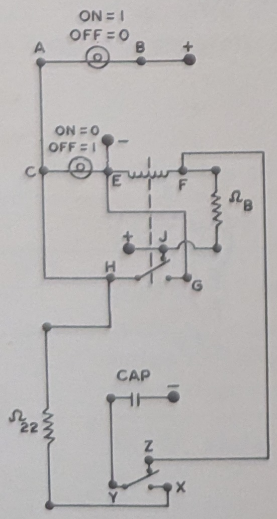
THIS FLIP-FLOP CAN BE BUILT ON ANY OF THE SIX SECTIONS OF THE MINIVAC.

This flip-flop uses a biased relay as a bi-stable element. A capacitor provides the energy surge to turn the relay on; the capacitor is also used as a storage element for the relay coil energy when the relay is to be turned off. The advantage of this single-relay flip-flop is the speed at which it may be operated.

The state of this flip-flop is indicated by the binary output light and the relay indicator light. As in the previous experiment, the two lights are complementary:

Binary Output Light	ON = 1	OFF = 0
Relay Indicator Light	ON = 0	OFF = 1

The program and circuit diagram for a fast flip-flop are:



A/C	$\begin{pmatrix} F/\Omega_B \\ \Omega_B/J \end{pmatrix}$
B/+	
C/H	$\begin{pmatrix} H/\Omega_{22} \\ \Omega_{22}/X \end{pmatrix}$
E/−	
E/G	J/+
F/Z	Y/cap

Procedure:

1. Wire this program using the appropriate resistor for the relay (see page 9). Turn power on. The relay indicator light comes on and the binary light remains off: the flip-flop is now storing "zero" (0).
2. Push the pushbutton and hold it down. The capacitor is now charging through the 22 ohm resistor.
3. Release the pushbutton. The energy which was stored in the capacitor was transferred to the relay coil, pulling the relay arm in. The relay remains on after this initial surge because the voltage across the biased relay coil is above the "drop-out" voltage. The lights indicate that the flip-flop is storing "one" (1).
4. Push the pushbutton and hold it down. The capacitor is now discharging through the 22 ohm resistor.
5. Release the pushbutton. The energy in the relay coil was momentarily directed to the discharged capacitor. The voltage across the relay coil went below the "drop-out" voltage and the relay arm dropped out. The lights now indicate that the flip-flop is storing "zero" (0).
6. Now rapidly push and release the pushbutton several times. Notice that, unlike the flip-flop in Experiment 1, this flip-flop does not miss when operated at high speed.

In this circuit the capacitor discharges through the 22 ohm resistor; in the previous circuit, the capacitor discharged through the resistance provided by the light. Discharge through the lower-valued resistance (22 ohms) requires less time than discharge through the light, so the flip-flop can operate at higher speeds.

This flip-flop may be combined with an oscillator for automatic operation. (See Experiment 3 which follows).

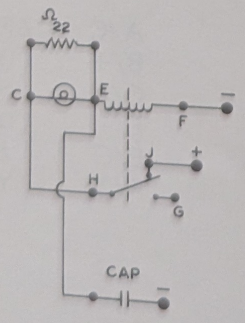
Experiment 3: Oscillator

THIS CIRCUIT MAY BE BUILT ON ANY OF THE SIX SECTIONS OF THE MINIVAC.

The oscillator is the basic timing element for computer operations. Full-scale computer systems use an electronic oscillator which is identical in performance to this relay oscillator. The pulses produced by an oscillator establish timing for the computer's operations. (The electronic oscillator in a computer system is frequently referred to as the "clock".)

If built on section 6, the oscillator will operate more slowly because of the larger capacitor in section 6. The speed at which the oscillator operates (called the "oscillation frequency") may also be varied by changing the values of the resistors used in the circuit.

The program and circuit diagram for an oscillator are:



C/H
$\begin{pmatrix} C/\Omega_{22} \\ \Omega_{22}/E \end{pmatrix}$
E/cap
F/−
J/+

Procedure:

1. Turn power on. The oscillator immediately begins to operate. When the relay is off, the capacitor begins to charge through the normally closed relay contact and the resistance provided by the light and the resistor. When the voltage in the capacitor reaches the relay's "pull-in" voltage the relay pulls in, disconnecting the "charging" circuit. At this point, the energy from the capacitor is going through the relay coil; that is, the capacitor is discharging through the relay coil. When the voltage in the capacitor reaches the relay's "drop-out" voltage the relay drops out, disconnecting the "discharge" circuit. The cycle then begins again.
2. The speed at which the oscillator operates (the oscillation frequency) may be changed by changing the value of the resistor between C and E. Notice that the oscillator will operate more slowly with higher resistance.
3. The oscillation frequency may be slowed further by connecting a resistor between G and J. Try successively smaller values of resistance until the oscillator stops.

Operating a Fast Flip-Flop with an Oscillator

The oscillator may be used to automatically operate the fast flip-flop. With the oscillator wired on one section, build a fast flip-flop (see Experiment 2) on another section. Then transfer the pushbutton connections on the flip-flop to the free contacts on the oscillator's relay. That is, transfer:

X on flip-flop to K on oscillator

Y on flip-flop to L on oscillator

Z on flip-flop to N on oscillator

The oscillator pulses now automatically operate the flip-flop. The speed at which the flip-flop operates can be varied by changing the oscillation frequency.

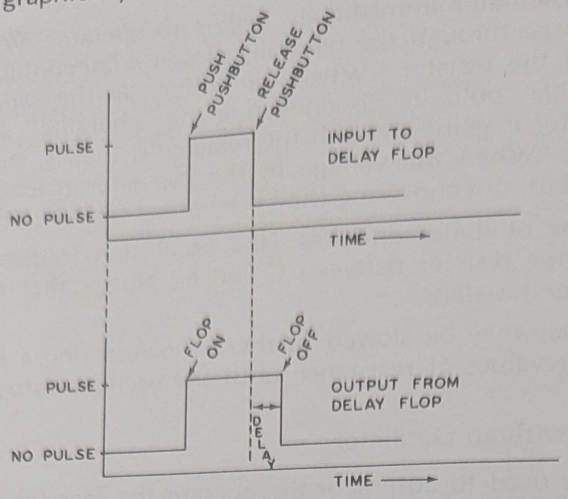
Experiment 4: Delay Flop — Pulse Input

THIS PROGRAM MAY BE BUILT ON ANY OF THE SIX SECTIONS OF THE MINIVAC

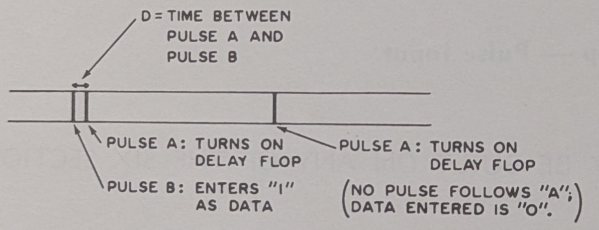
The "delay flop" (sometimes called the "one-shot multivibrator") is used in computer systems to provide a specified time delay between changes of state. A delay flop differs from a flip-flop in that it turns itself off. That is, once a delay flop is turned on, it will turn itself off after a given time; it does not require a separate pulse to turn it off. The chart below shows this difference:

	Delay Flop (Pulse input)	**Flip-Flop**
start	in "0" state	in "0" state
push pushbutton	goes to "1" state	goes to "1" state
release pushbutton	holds "1" state for specified time; then returns to "0" state	holds "1" state
finish	in "0" state	in "1" state

We can also show graphically the effect produced by a delay flop with pulse input:

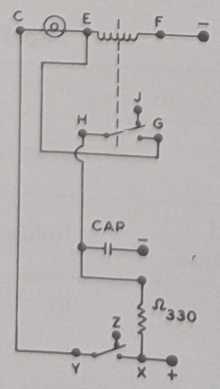


This delay flop can be used, for example, in the reading of data from magnetic tape. The magnetic marks on the tape provide input pulses to a delay flop. Each input pulse turns on the delay flop with the instruction "prepare to read data". The delay flop opens a switch to the correct channel for the data and holds it open for a specified length of time. If a second pulse follows immediately after the "prepare" pulse, the data being entered is "1". If, on the other hand, there is **no** second pulse within the delay time, the data entered is "0". We can show this as:


D = TIME BETWEEN PULSE A AND PULSE B
PULSE A: TURNS ON DELAY FLOP
PULSE B: ENTERS "1" AS DATA
PULSE A: TURNS ON DELAY FLOP
(NO PULSE FOLLOWS "A"; DATA ENTERED IS "0".)

The flop used in this example would have a delay greater than d — the time between pulses A and B. Any "1" input (that is, a "B" pulse) would follow an "A" pulse within the time the delay flop was on.

The program below is for a delay flop with "pulse" input. In the experiment which follows, we will examine a delay flop with "continuous" input.



C/Y
E/G
F/−
H/cap
$\begin{pmatrix} \text{cap}/\Omega_{330} \\ \Omega_{330}/\text{X} \end{pmatrix}$*
X/+

*Try various values of resistance for best operation.

Procedure:

1. Turn power on. The flop is now in the "0" state. The relay is off.
2. Push the pushbutton and hold it down. The flop is holding in the "1" state (the relay stays on).
3. Release the pushbutton. The flop will remain on for a short time after the pushbutton has been released. Then it will turn itself off. (The relay off indicates that the flop is off.)
4. Now make the following changes to the program:

change

C/Y	to	C/D16
Ω_{330}/X	to	Ω_{330}/D13
X/+	to	D13/+

add

D17/M+
D18/M−

5. Turn power on. The rotary switch dial begins to turn. When it passes D13, it produces an input pulse — the same effect as pushing and releasing the pushbutton. Notice that although the flop turns on as the dial passes D13, the flop does not immediately turn off. The short pulse generated by the rotary switch has been delayed.

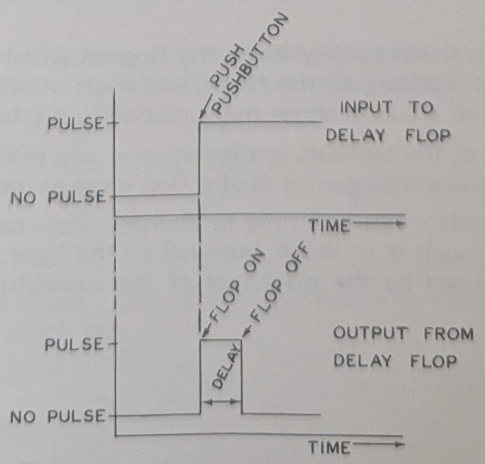
Note: to increase the delay time, connect the flop's capacitor in parallel with other capacitors. For example, if your delay flop is built on section 1 of the MINIVAC, you can increase its delay time by making the connections:

1cap/2cap
2cap/3cap
etc.

Experiment 5: Delay Flop — Continuous Input

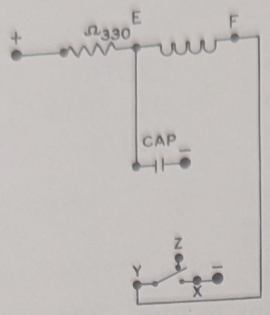
THIS PROGRAM MAY BE BUILT ON ANY OF THE SIX SECTIONS OF THE MINIVAC

This delay flop requires a "continuous" input which is at least as long as the flop's delay time. That is, the pushbutton must be held down at least as long as the delay time. After that time, the time at which the pushbutton is released is not important. We can show this graphically as:



As this graph shows, if the input pulse is at least as long as the flop's delay time, the flop will automatically turn off at the end of the delay.

A delay flop of this type is frequently used to control another circuit. A pulse into the flop produces an output pulse at a specified, later time. It is also used as a "logical gate" which allows signals to pass (or prevents their passing) during the flop's delay time. The continuous input delay flop, like the pulse input delay flop, is useful when a delay is required for proper timing.



E/cap
(E/Ω330 / Ω330/+)
F/Y
X/−

Note: If this program does not operate to your satisfaction, try one of the following:

1. Substitute a 270 ohm resistor for the 330 ohm resistor in the circuit.
2. Use the 330 ohm resistor; change "E" connections to "C":

change E/cap	to	C/cap
change E/Ω330	to	C/Ω330

3. Use a 270 ohm resistor and change "E" connections to "C":

change E/cap	to	C/cap
change (E/Ω330 / Ω330/+)	to	(C/Ω270 / Ω270/+)

Procedure:

1. Push the pushbutton and hold it down. The flop will come on, remain on for a short time and then turn itself off.
2. Wait long enough for the capacitor to charge. Then quickly push and release the pushbutton. If the input pulse is shorter than the flop's delay time, the flop will turn off when the pulse is removed.

The energy stored in the capacitor turns the flop on when the pushbutton is pushed. As the energy in the capacitor is discharged, the flop turns itself off when the voltage falls to the relay's drop-out voltage. We can visually show the capacitor's discharge as follows:

3. Select a light on one of the sections not being used and make the connection B/−. Now connect one end of a programming wire to the flop's capacitor.
4. Be sure that the capacitor has had time to charge. Now take the free end of the wire to the flop's capacitor and touch it to the A terminal of the light. The light will come on and then fade out. This is caused by the discharge of the capacitor.

Experiment 6: Six-Bit Binary Counter

This experiment illustrates how flip-flops may be combined to form a binary counter. Since the fast flip-flop (see Experiment 2) uses only one relay, we will be able to connect six flip-flops to build a six-bit counter. (See Book IV, page 71 for a 3-bit counter).

This is the program for a fast flip-flop from experiment 2:

A/C	$\begin{pmatrix} F/\Omega_B \\ \Omega_B/J \end{pmatrix}$
B/+	
C/H	$\begin{pmatrix} H/\Omega_{22} \\ \Omega_{22}/X \end{pmatrix}$
E/−	
E/G	J/+
F/Z	Y/cap

To build a six-bit binary counter:

1. Build a fast flip-flop on ***each*** of the six sections of the MINIVAC.
2. Now transfer the pushbutton connections on sections 1 through 5 to the contacts of the relay to the right.
   That is, change:

1X	to 2K	3Z	to 4N
1Y	to 2L	4X	to 5K
1Z	to 2N	4Y	to 5L
2X	to 3K	4Z	to 5N
2Y	to 3L	5X	to 6K
2Z	to 3N	5Y	to 6L
3X	to 4K	5Z	to 6N
3Y	to 4L		

Pushbutton 6 will now operate the counter. This pushbutton controls the section 6 flip-flop which in turn controls the section 5 flip-flop. The section 5 flip-flop controls the section 4 flip-flop, and so on.

For convenience in reading the counter's results, label each output light with its appropriate weight:

Light	1	2	3	4	5	6
Weight	32	16	8	4	2	1
	(2^5)	(2^4)	(2^3)	(2^2)	(2^1)	(2^0)

Procedure:

1. Turn power on. All output lights are off, indicating that the counter is at zero (0).
2. Push pushbutton 6 and release it. The lights are now displaying:

Light	1	2	3	4	5	6
	(32)	(16)	(8)	(4)	(2)	1
	OFF	OFF	OFF	OFF	OFF	ON

   The counter is at one (1).

3. Push pushbutton 6 several times. The counter continues to count up in binary. The limit of this six-bit counter is 63, which is:

Light:	1	2	3	4	5	6
	(32)	(16)	(8)	(4)	(2)	(1)
	ON	ON	ON	ON	ON	ON
	1	1	1	1	1	1

Note: the decimal value of the number displayed in binary is calculated by adding the weights of those lights which are on. For example, the binary display:

Light	1	2	3	4	5	6
	(32)	(16)	(8)	(4)	(2)	(1)
	ON	OFF	ON	ON	OFF	ON

in decimal is: $32 + 8 + 4 + 1 = 45$

(If the counter does not operate to your satisfaction, try using different values for the bias resistors.)

Leave this circuit on the board for the experiment which follows.

Experiment 7: Automatic Five-Bit Binary Counter

This experiment uses the oscillator in Experiment 3 to operate a five-bit binary counter. The counter itself is five fast flip-flops connected together as in the previous experiment. Since the counter has only five flip-flops its limit is, in decimal, 31.

The operation of this automatic counter illustrates how a computer can count by itself, using an oscillator for timing. Automatic counters are the basis of most arithmetic circuits used in full-scale computers.

To build an automatic counter:

1. Program a six-bit binary counter (see Experiment 6).
2. With the six-bit counter on the MINIVAC console, remove the flip-flop on section 6.

   That is, remove:

   6A/6C
   6B/6+
   6C/6H
   6E/6−
   6E/6G
   6F/6Z
   (6F/Ω_{B6} or Ω_{B6}/6J)
   (6H/Ω_{22} or Ω_{22}/6X)
   6J/6+
   6Y/6cap

3. Now build an oscillator (see Experiment 3) on section 6. That is, add:

   6C/6H
   (6C/Ω_{22} or Ω_{22}/6E)
   6E/6cap
   6F/6−
   6J/6+

For convenience, label the output lights on sections 1 through 5 with the binary weights as follows:

Light	1	2	3	4	5
Weight	16	8	4	2	1
	(2^4)	(2^3)	(2^2)	(2^1)	(2^0)

Procedure:

1. Turn power on. The counter immediately begins to count. Notice that when the counter reaches its limit (31), it automatically returns to zero and begins counting again.

   Full-scale computers are frequently programmed to generate a control signal when a specified count is reached. In some cases, ***each*** count represents a different operation. We can illustrate this control function by the following:

2. Change the program for the automatic counter:

remove	1cap/2L	(5H/Ω_{22}
	5F/6N	Ω_{22}/6K)
		6J/6+
add	1A/1L	(5H/Ω_{22}
	1K/D18	Ω_{22}/6N)
	1N/6J	D15/M−
	1cap/D16	D17/M+
	2L/D0	(D18/Ω
	5F/6K	Ω/D19)*

** This is an optional connection. Use a 15, 22 or 33 ohm resistor to decrease the speed of the motor.*

3. Set the rotary switch at 0 and turn power on. When the counter reaches 10000 (16), the counter stops and the rotary switch makes a full revolution. When the rotary switch has returned to 0 it stops and the counter begins counting again from zero.

Note: the full program for the automatic counter with rotary switch control is:

1A/1C	2E/2G	(3H/Ω_{22}	5C/5H
1A/1L	2E/2−	Ω_{22}/4K)	5E/5G
1B/1+	2F/3N	3J/3+	5E/5−
1C/1H	(2F/Ω_{B2}	3cap/4L	5F/6K
1E/1G	Ω_{B2}/2J)	4A/4C	(5F/Ω_{B5}
1E/1−	(2H/Ω_{22}	4B/4+	Ω_{B5}/5J)
1F/2N	Ω_{22}/3K)	4C/4H	(5H/Ω_{22}
(1F/Ω_{B1}	2J/2+	4E/4G	Ω_{22}/6N)
Ω_{B1}/1J)	2L/DO	4E/4−	5J/5+
(1H/Ω_{22}	2cap/3L	4F/5N	5cap/6L
Ω_{22}/2K)	3A/3C	(4F/Ω_{B4}	6C/6H
1J/1+	3B/3+	Ω_{B4}/4J)	(6C/Ω_{22}
1K/D18	3C/3H	(4H/Ω_{22}	Ω_{22}/6E)
1N/6J	3E/3G	Ω_{22}/5K)	6F/6−
1cap/D16	3E/3−	4J/4+	D15/M−
2A/2C	3F/4N	4cap/5L	D17/M+
2B/2+	(3F/Ω_{B3}	5A/5C	(D18/Ω
2C/2H	Ω_{B3}/3J)	5B/5+	Ω/D19)*

** Optional*

Experiment 8: Counters for Bases 9, 10, 11, 12, 13, 14, 15 and 16.

The counters in the previous experiments display numbers according to a specific ***binary code.*** This is known as the "8421 code" and represents the weight of the four low-order bits of the counter. By using an appropriate code — that is, by weighing each bit correctly — we can display numbers in bases other than decimal or binary.

As we pointed out in Book IV, it is possible to count from zero using a number other than ten as a base (see Book IV, page 67, ff.). When we count to a base greater than ten, we must use a single character to represent each number up to the number which we are using as a base. For convenience, we use alphabetic characters for this:

Counting from 0 to $(32)_{10}$ in Various Bases

Base	9	10	11	12	13	14	15	16
0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8	8
10	9	9	9	9	9	9	9	9
11	10	A	A	A	A	A	A	A
12	11	10	B	B	B	B	B	B
13	12	11	10	C	C	C	C	C
14	13	12	11	10	D	D	D	D
15	14	13	12	11	10	E	E	E
16	15	14	13	12	11	10	F	F
17	16	15	14	13	12	11	10	10
18	17	16	15	14	13	12	11	11
20	18	17	16	15	14	13	12	12
21	19	18	17	16	15	14	13	13
22	20	19	18	17	16	15	14	14
23	21	1A	19	18	17	16	15	15
24	22	20	1A	19	18	17	16	16
25	23	21	1B	1A	19	18	17	17
26	24	22	20	1B	1A	19	18	18
27	25	23	21	1C	1B	1A	19	19
28	26	24	22	20	1C	1B	1A	1A
30	27	25	23	21	1D	1C	1B	1B
31	28	26	24	22	20	1D	1C	1C
32	29	27	25	23	21	1E	1D	1D
33	30	28	26	24	22	20	1E	1E
34	31	29	27	25	23	21	1F	1F
35	32	2A	28	26	24	22	20	20

To count to these bases on the MINIVAC, we will start with a basic, four flip-flop counter on relays 3, 4, 5 and 6. To this we will add a modified delay flop on relay 2. The delay flop will be used to re-set certain flip-flops so that we can select the appropriate weighting for each bit of the counter.

The weightings for each bit of the counter (that is, the "code") for the bases in the table above are:

Base	Code (Weighing)
16	8421
15	7421
14	6421
13	5421
12	4421
11	3421
10	2421
9	1421

The code gives the weight of each bit. For example, if the counter is programmed for base 13, lights 3 through 6 would represent:

Light	3	4	5	6
Weight	5	4	2	1

and, if the lights displayed:

Light	3	4	5	6
Weight	(5)	(4)	(2)	(1)
	ON	OFF	ON	ON

the number 5 + 2 + 1 = 8 would be displayed.

The basic program for all counters is:

2A/2−
2B/2N
2E/2cap
(2E/Ω_{330} Ω_{330}/2+)
(2F/Ω_{15} Ω_{15}/3com)
2H/2+
2H/2L
3A/3C
3B/3+
3C/3H
3E/3−
3E/3G
3F/4N
(3F/Ω_{B3} Ω_{B3}/3J)
3H/3com
3J/3+
(3com/Ω_{22} Ω_{22}/4K)
3cap/4L
4A/4C
4B/4+
4C/4H
4E/4−
4E/4G
4F/5N
(4F/Ω_{B4} Ω_{B4}/4J)
(4H/Ω_{22} Ω_{22}/5K)
4J/4+
4cap/5L
5A/5C
5B/5+
5C/5H
5E/5−
5E/5G
5F/6N
(5F/Ω_{B5} Ω_{B5}/5J)
(5H/Ω_{22} Ω_{22}/6K)
5J/5+
5cap/6L
6A/6C
6B/6+
6C/6H
6E/6−
6E/6G
6F/6Z
(6F/Ω_{B6} Ω_{B6}/6J)
(6H/Ω_{22} Ω_{22}/6X)
6J/6+
6Y/6cap

This program produces a four-bit counter which will count to 16 using the code 8421. The chart which follows shows the program additions which will permit the counter to operate in different bases:

Base	*Code*	*Add these connections to the basic program*
16	8421	none
15	7421	2G/6Z
14	6421	2G/6N
13	5421	2G/6N, 2K/6Z
12	4421	2G/5N
11	3421	2G/5N, 2K/6Z
10	2421	2G/5N, 2K/6N
9	1421	1E/1−, 1F/2G, 1G/6Z, 1H/1L, 1H/2L, 1K/6N, 2K/5N

Procedure:

1. Wire the basic program onto the MINIVAC. Now select the base to which you would like to count, and add the proper connections (see table above).
2. Turn power on. Push pushbutton 6 until the delay flop turns off. (Relay 2 will go off.) The counter is now at zero and is ready to operate. (Note: whenever the counter is re-set, push pushbutton 6 until the delay flop turns off and the counter is at zero.)
3. Push pushbutton 6 several times. The counter's results will be displayed on lights 3 through 6 according to the appropriate code. (Ignore light 2.)

For example, to count to the base 13:

To the basic program, add the connections 2G/6N and 2K/6Z

For convenience, label lights 3 through 6 with the code:

Light	3	4	5	6
Weight	5	4	2	1

Push pushbutton 6 several times. Using the notation that a light on represents a 1 and a light off represents 0, the following sequence will be displayed:

decimal count	Lights Code 3 5	4 4	5 2	6 1		count to base 13
0	0	0	0	0	=	0
1	0	0	0	1	=	1
2	0	0	1	0	=	2
3	0	0	1	1	=	3
4	0	1	0	0	=	4
5	0	1	0	1	=	5
6	0	1	1	0	=	6
7	0	1	1	1	=	7
8	1	0	1	1	=	8
9	1	1	0	0	=	9
10	1	1	0	1	=	A
11	1	1	1	0	=	B
12	1	1	1	1	=	C

Note: If the program does not operate to your satisfaction, check all bias resistors in the circuit. This program may require different bias resistances for some relays.

Experiment 9: Decimal to Binary Converter

The program for a decimal to binary converter given in Book IV uses a ***decoding*** circuit. That is, the computer is programmed to automatically translate from decimal into binary. The program which follows uses a ***counting*** circuit for conversion. (The "decoding" converter is in Book IV, page 84 ff.)

The first five sections of the MINIVAC are programmed as a binary counter (8421 code). Connections are then made from the rotary switch through relay 6 to the counter. As the rotary switch turns, it makes contact with the terminals and produces pulses which operates the counter. The number displayed in binary on lights 1 through 5 is the result of the number of pulses entered on the rotary switch.

The program for a "counting" converter is:

1A/1+	2J/2+	$\begin{pmatrix}4H/\Omega_{22}\\ \Omega_{22}/5K\end{pmatrix}$	D0/D19
1B/1C	2cap/3L	4J/4+	D1/D2
1C/1H	3A/3+	4cap/5L	D2/D3
1E/1−	3B/3C	5A/5+	D3/D4
1E/1G	3C/3H	5B/5C	D4/D5
1F/2N	3E/3−	5C/5H	D5/D6
$\begin{pmatrix}1F/\Omega_{B1}\\ \Omega_{B1}/1J\end{pmatrix}$	3E/3G	5E/5−	D6/D7
$\begin{pmatrix}1H/\Omega_{22}\\ \Omega_{22}/2K\end{pmatrix}$	3F/4N	5E/5G	D7/D8
1J/1+	$\begin{pmatrix}3F/\Omega_{B3}\\ \Omega_{B3}/3J\end{pmatrix}$	5F/6N	D8/D9
1cap/2L	$\begin{pmatrix}3H/\Omega_{22}\\ \Omega_{22}/4K\end{pmatrix}$	$\begin{pmatrix}5F/\Omega_{B5}\\ \Omega_{B5}/5J\end{pmatrix}$	D9/D10
2A/2+	3J/3+	$\begin{pmatrix}5H/\Omega_{22}\\ \Omega_{22}/6K\end{pmatrix}$	D10/D11
2B/2C	3cap/4L	5J/5+	D11/D12
2C/2H	4A/4+	5cap/6L	D12/D13
2E/2−	4B/4C	6C/D1	D13/D14
2E/2G	4C/4H	$\begin{pmatrix}6C/\Omega_{150}\\ \Omega_{150}/6E\end{pmatrix}$	D14/D15
2F/3N	4E/4−	6F/6−	D16/M+
$\begin{pmatrix}2F/\Omega_{B2}\\ \Omega_{B2}/2J\end{pmatrix}$	4E/4G		D16/D18
$\begin{pmatrix}2H/\Omega_{22}\\ \Omega_{22}/3K\end{pmatrix}$	4F/5N		D17/M−
	$\begin{pmatrix}4F/\Omega_{B4}\\ \Omega_{B4}/4J\end{pmatrix}$		$\begin{pmatrix}D18/\Omega_{33}\\ \Omega_{33}/D19\end{pmatrix}$

Procedure:

1. Set the rotary switch dial to any number from 0 through 15.
2. Turn power on. The rotary switch returns to zero, sending a pulse to the counter as it passes each rotary switch terminal. The decimal number entered will be displayed in binary on lights 1 through 5 when the pointer knob has stopped at zero.
3. To convert another number, turn power off and set the dial. Then turn power on again.

This program can easily be changed to produce an adder which accepts decimal input and displays its results in binary:

4. Change D16/M+ to D16/6V; add the connection 6U/6+. The converter can now be operated as an adder.
5. Set slide switch 6 RIGHT and turn power on. Set the rotary switch dial to a decimal number from 0 through 15. Set slide switch 6 LEFT. The rotary switch returns to zero and the decimal number entered is displayed in binary on lights 1 through 5.
6. Set slide switch 6 RIGHT and enter the number to be added by setting the rotary switch dial. Set slide switch 6 LEFT. The rotary switch returns to zero and the number displayed in binary is the **sum** of the decimal numbers entered. (Note: the limit of this adder is 31.)
7. To reset the adder, turn power off, then on again.

Leave this program on the MINIVAC for the experiment which follows.

Experiment 10: Multiplication with a Counter

This experiment demonstrates how a computer multiplies using a counter. Some full-scale computors multiply in this fashion, although it is a relatively slow method. High-speed computers generally employ an accumulator to carry out multiplications; this reduces computation time but requires more equipment than the counter method.

To build a counter for multiplication:
Build the decimal to binary converter from Experiment 9 and modify it for an adder. Then make the following changes:

remove	6U/6+
	6V/D16
	D0/D19
add	6X/6+
	6Y/D16

Procedure:

1. Select two numbers to be multiplied. (Their product should not exceed 31.) Enter the larger number by setting the rotary switch dial to that digit.
2. Remove the connection on the rotary switch from the number which has been entered to the next higher number. (For example, if 7 has been entered, remove the connection D7/D8.)
3. Turn power on. Push pushbutton 6 down and hold it down while the rotary switch makes as many revolutions on the multiplier. (For example, if the multiplier is 3, let the rotary switch make 3 complete revolutions.)
4. The number displayed in binary on lights 1 through 5 is the product of the numbers multiplied.
5. To multiply another set of numbers, replace the connection on the rotary switch and turn power off.

As an example, multiply 4 times 6:

Set the rotary switch to 6.
Remove D6/D7.
Turn power on.
Push pushbutton 6 and hold it down while the rotary switch makes 4 complete revolutions.
The product is displayed in binary:

11000 = 24

Full-scale computers use a separate counter to keep track of the revolutions and stop the motor automatically when the multiplier is reached. This step can be simulated on the MINIVAC by using a second MINIVAC to count the revolutions of the rotary switch and stop the motor automatically.

Leave this program wired on the MINIVAC for the experiment which follows.

Experiment 11: Division with a Counter

Division can be carried out on a computer by using a ***down counter*** just as an ***up counter*** was used to perform multiplications. The dividend is entered on the down counter and the divisor is entered on the rotary switch. As the rotary switch turns, the counter counts ***down*** (towards zero). The quotient is the number of revolutions which the rotary switch completed before the remainder (displayed on the counter) is less than the divisor.

The counter used in Experiment 10 for multiplication is actually five flip-flops connected together. Referring to Experiment 2 (Fast Flip-Flop) we see that with this flip-flop the binary output light ***complements*** the relay indicator light. If we turn on the power with the program from Experiment 10 wired, binary output lights 1 through 5 will show:

0 0 0 0 0 (all lights off)

However, relay indicator lights 1 through 5 show the complement of 0 0 0 0 0, or:

1 1 1 1 1 (all lights on)

When we push pushbutton 6, the binary output lights display the count ***up*** from zero. At the same time, the relay indicator lights are displaying the complement of each count; they are, in fact, displaying the count ***down*** from 31.

We can therefore use the same program for division that we used for multiplication by observing the ***relay indicator lights*** which are acting as a down counter.

Procedure:

1. With the program from Experiment 10 wired, set the rotary switch to 0 and turn power on. The number displayed in the down counter (on the relay indicator lights) is 11111 or 31. To enter the dividend into the counter, we must turn off the appropriate relays. This is done by plugging one end of a programming wire into M+; the free end of this wire is then touched to the F terminal of the relays which are to be turned off.
2. The divisor is now entered on the rotary switch by removing the connection from the divisor's digit to the next higher digit. (For example, if the divisor is 7, remove D7/D8.)
3. The division is carried out by pushing pushbutton 6 and holding it down. The quotient is the number of complete revolutions made by the rotary switch before the number displayed in the down counter is less than the divisor. The number finally displayed in the down counter is the remainder.

As an example, we will divide 22 by 5:

Enter 22 (10110) in the down counter by touching the wire from M+ to 2F and 6F. (This turns off relays 2 and 5.)
Enter 5 (00101) on the rotary switch by removing D5/D6.

Push pushbutton 6 and hold it down until the number displayed in the down counter (on the relay indicator lights) is less than the divisor (less than 00101).

The rotary switch makes 4 complete revolutions before the number in the down counter becomes less than the divisor. After 4 revolutions, the down counter displays 00010 or 2.

Therefore:

$22 \div 5 = 4$ and remainder 2

To divide another set of numbers, turn power off and replace the connection on the rotary switch.

The programs in this book depend upon proper operation of the relays. If you are experiencing program difficulties, check each relay. (See page 7 of the Maintenance Manual.)

The selection of resistors for a program is critical. Always be sure that the bias resistances are correct ***after*** a program has been wired. (Check a bias resistance by testing whether the relay will remain both on and off when power is on.)

Bias resistances are correct if all relays used in a program will remain on at the same time.

If a program does not use all six capacitors, its operation may be improved by connecting a 33 ohm resistor from + power to an unused capacitor.

The values used for resistors other than bias resistors are correct for most cases. Slight variations in individual components may require that you change these resistance values for satisfactory operation of a particular program.

Experiment 12. Combined Arithmetic Program

The preceding arithmetic programs may be combined into a single, general program which will allow you to perform six different experiments with one circuit. By setting the slide switches appropriately, the following program can be used as:

1. Five-bit binary counter operated by the rotary switch
2. Five-bit binary counter operated by an oscillator (see Experiment 7)
3. Five-bit binary counter controlling rotary switch (see Experiment 7)
4. Decimal to Binary Converter (see Experiment 9)
5. "Counter" method Mutliplier (see Experiment 10)
6. "Counter" method Divider (see Experiment 11)

The combined arithmetic program is:

1A/1+
1A/1L
1B/1C
1C/1H
1E/1G
1E/1−
1F/2N
(1F/Ω_{B1} Ω_{B1}/1J)
(1H/Ω_{22} Ω_{22}/2K)
1J/1+
1K/4R
1N/3R
1R/2V
1S/D0
1U/1X
1U/D17
1V/4S
1W/1Y
1cap/2S
2A/2+
2B/2C
2C/2H
2E/2−
2E/2G
(2F/Ω_{B2} Ω_{B2}/2J)
2F/3N
(2H/Ω_{22} Ω_{22}/3K)
2J/2+
2L/2T
2R/4U
2T/2U
2W/D19
2cap/3L
3A/3+
3B/3C
3C/3H
3E/3−
3E/3G
(3F/Ω_{B3} Ω_{B3}/3J)
3F/4N
(3H/Ω_{22} Ω_{22}/4K)
3J/3+
3S/5T
3U/6cap
3V/6E
3W/D1
3cap/4L
4A/4+
4A/4W
4B/4C
4C/4H
4C/4G
4E/4−
(4F/Ω_{B4} Ω_{B4}/4J)
4F/5N
(4H/Ω_{22} Ω_{22}/5K)
4J/4+
4T/4−
4T/6R
4V/D16
4cap/5L
5A/5+
5B/5C
5C/5H
5E/5−
5E/5G
(5F/Ω_{B5} Ω_{B5}/5J)
5F/6N
(5H/Ω_{22} Ω_{22}/6K)
5J/5+
5R/6+
5S/6J
5cap/6L
6C/6H
(6C/Ω_{150} Ω_{150}/6E)
6F/6−
(6G/Ω_{330} Ω_{330}/6J)
6R/6V
6S/D18
6T/D16
6U/D15
D1/D2
D2/D3
D3/D4
D4/D5
D5/D6
D6/D7
D7/D8
D8/D9
D9/D10
(D18/Ω_{22} Ω_{22}/D19)

Procedure:

The slide switch settings for the various operations are:

	Slide Switch					
	1	2	3	4	5	6
1. Counter (rotary switch operated)	Left	Right	Right	Right	Right	Right
2. Counter (oscillator operated)	Left	Right	Left	Left	Left	Right
3. Counter Controlling rotary switch	Left	Left	Left	Left	Right	Left
4. Decimal to binary converter	Left	Right	Right	Right	Right	Right
5. "Counter Method" Multiplier	Right	Right	Right	Right	Right	Right
6. "Counter Method" Divider	Right	Right	Right	Right	Right	Right

To re-set the computer between programs turn power off, then on again.

1. Counter (rotary switch operated):

Remove the connections 1S/D0 and D1/D2. Set the slide switches (see above) and turn power on. The counter automatically begins to count up from zero. Output lights 1 through 5 display the count in binary. (The counter's limit is decimal 31.)

The count pulses to the counter are generated by the rotary switch. The counter receives a pulse each time the rotary switch passes 1.

Replace the connections 1S/D0 and D1/D2 for the operations which follow.

2. Counter (oscillator operated):
   Set the slide switches (see above) and turn power on. The counter automatically begins to count up from zero. Output lights 1 through 5 display the count in binary. The counter is operated by the pulses generated by the oscillator on section 6.

3. Counter controlling rotary switch:
   Set the slide switches (see above) and set the rotary switch at 0. Turn power on. The counter automatically counts to 17; when it reaches 17, the counter stops and the rotary switch makes a full revolution. When the rotary switch stops, the counter begins counting from zero again and the cycle is repeated.

4. Decimal to binary converter:
   Set the slide switches (see above). Enter a decimal number from 0 through 10 on the rotary switch. Turn power on.
   The rotary switch returns to zero and output lights 1 through 5 display the binary equivalent of the decimal number entered.

5. "Counter method" multiplier:
   Set the slide switches. Select two numbers to be multiplied together (the limit of the product is 31.) Enter the larger number on the rotary switch by ***removing*** the rotary switch connection from the number to the next higher number. (For example, to enter 8 remove D8/D9.)
   Turn power on. Hold pushbutton 1 down while the rotary switch makes as many revolutions as the multiplier. (For example, if the number in the rotary switch is to be multiplied by 4, hold pushbutton 1 down while the rotary switch makes four complete revolutions.) The product will be displayed in binary on lights 1 through 5. Replace the connection on the rotary switch before proceeding.

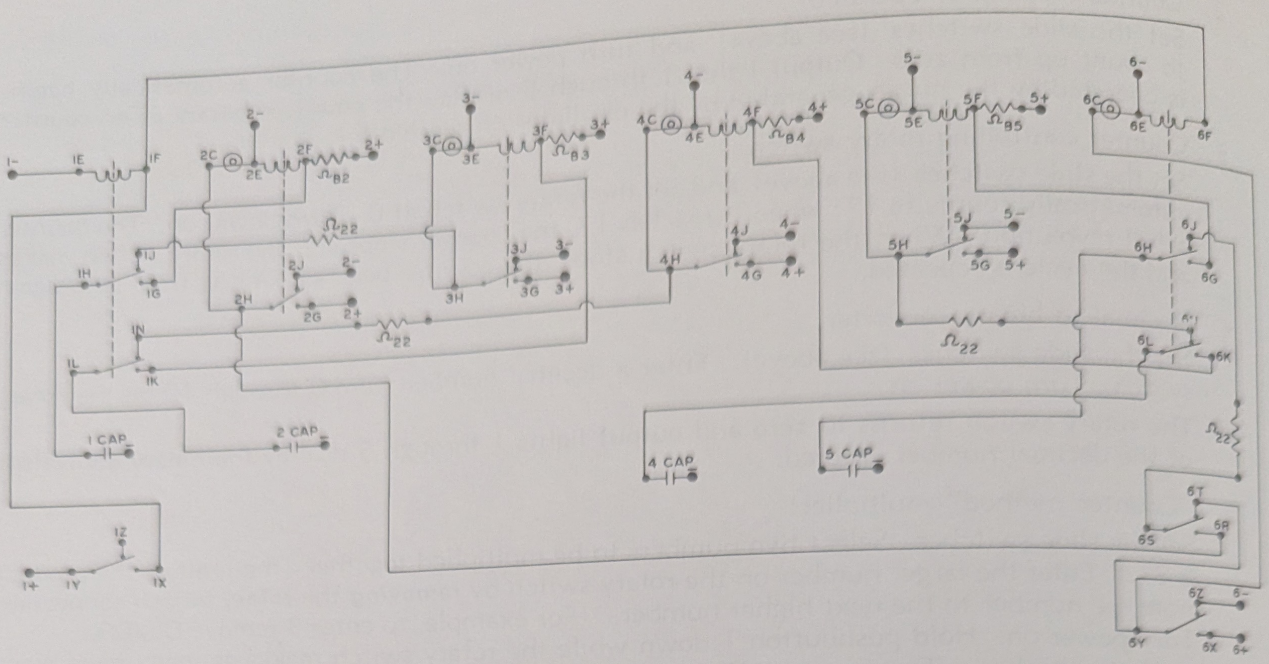
6. "Counter method" divider:
   Set the slide switches (see above) and turn power on. Enter the dividend in binary on ***relay indicator lights*** 1 through 5. (Turn a light off by touching a wire from M+ to the F terminal of the relay. Enter the number from ***right*** to ***left***.)
   Enter the divisor on the rotary switch by removing the rotary switch connection from the number to the next higher number. (For example, to enter 5, remove D5/D6.)
   Push pushbutton 1 and hold it down while the rotary switch turns. The quotient is the number of complete revolutions the rotary switch makes before the number displayed on the relay indicator lights is less than the divisor. The number displayed on the relay indicator lights is the remainder.

Experiment 13: Four-Bit Automatic Shift Register

The capacitors and resistors allow us to build a four-bit shift register which will automatically shift left one bit each time a pushbutton is pushed. (For other shift registers see Book IV, page 80 and page 86.) When the shift command is given, each stage of the shift register transfers its information to the next stage to the left.

The circuit program and diagram for the shift register are:

1E/1−
1F/1X
1F/6F
1G/2F
1H/1 cap
(1J/Ω_{22} Ω_{22}/3H)
1K/3F
1L/2 cap
1N/4 com
1Y/1+
2C/2H
2E/2−
(2F/Ω_{B2} Ω_{B2}/2+)
2G/2+
2H/6R
2J/2−
3C/3H
3E/3−
(3F/Ω_{B3} Ω_{B3}/3+)
3G/3+
3J/3−
4C/4H
4E/4−
4F/6K
(4F/Ω_{B4} Ω_{B4}/4+)
4G/4+
(4H/Ω_{22} Ω_{22}/4 com)
4J/4−
4 cap/6L
5C/5H
5E/5−
5F/6G
(5F/Ω_{B5} Ω_{B5}/5+)
5G/5+
(5H/Ω_{22} Ω_{22}/6N)
5J/5−
5 cap/6H
6C/6Y
6E/6−
(6J/Ω_{22} Ω_{22}/6S)
6T/6Y
6X/6+
6Z/6−



Procedure:

1. Set slide switch 6 RIGHT and turn power on.
2. Pushbutton 6 is used to enter "data" into the shift register. To enter zero (0) leave pushbutton 6 up; to enter one (1) push pushbutton 6 and hold it down while pushing pushbutton 1.
3. Pushbutton 1 gives the "shift command." Each time pushbutton 1 is pushed, all data in the shift register is shifted one bit to the left.
4. Hold pushbutton 6 down and push pushbutton 1. Release both pushbuttons. Sections 2 through 4 are now displaying 0001 (on the relay indicator lights).
5. Push pushbutton 1 and release. The information in the shift register has shifted left and the register now displays 0010. (Input from pushbutton 6 was 0.)
6. Hold pushbutton 6 down and push pushbutton 1. Release both pushbuttons. The register now displays 0101.
7. Continue pushing pushbutton 1 until the information in the shift register has been shifted out. (The shift register is now "cleared.")

Shift registers may be "closed" so that information in the last stage is shifted back to the first stage rather than being shifted out. This shift register is designed to act as a closed shift register when slide switch 6 is LEFT.

8. Enter "1" into the shift register (push pushbutton 1 while holding down pushbutton 6). "Close" the shift register by moving slide switch 6 LEFT.
9. Push pushbutton 1 four times. Notice that the "1" is transferred from the last stage back to the first stage. (This is often called "end-around carry.")

The method of entering data bit by bit is known as entering data "in series." In some full-scale machines, data is entered into the shift register all at once; that is, data is entered to each stage "in parallel." Data may be shifted out of a register either "in series" or "in parallel." Such shift register operations are often referred to as "parallel in, serial out," "serial in, serial out," and so on.

Experiment 14: Multiplication with an Accumulator

In Book IV we explained how multiplications are carried out on full-scale computers. (See Book IV, page 79 ff.) An ***accumulator*** was used to perform the necessary shifts and additions.

With capacitors and resistors, we can now build an accumulator with automatic shifting capabilities. Multiplications will be carried out in binary as before, but we will no longer have to manually shift.

The program for an accumulator is:

1A/1+	2F/2S	3T/4K	$\begin{pmatrix}5F/\Omega_{B5}\\ \Omega_{B5}/5+\end{pmatrix}$
1A/1V	2G/3G	3W/4N	5F/5S
1B/1C	$\begin{pmatrix}2H/\Omega_{22}\\ \Omega_{22}/2V\end{pmatrix}$	3Y/4V	5G/6G
1C/1H	2J/3J	3 cap/4L	$\begin{pmatrix}5H/\Omega_{22}\\ \Omega_{22}/5V\end{pmatrix}$
1E/1−	2R/2W	4A/4+	5J/6J
$\begin{pmatrix}1F/\Omega_{B1}\\ \Omega_{B1}/1+\end{pmatrix}$	2S/2X	4B/4C	5R/5W
1F/2N	2T/2U	4C/4H	5S/5X
1G/1R	2T/3K	4E/4−	5T/5V
1G/2G	2W/3N	$\begin{pmatrix}4F/\Omega_{B4}\\ \Omega_{B4}/4\end{pmatrix}$	5T/6K
$\begin{pmatrix}1H/\Omega_{22}\\ \Omega_{22}/2K\end{pmatrix}$	2Y/3V	4F/4S	5W/6N
1J/1U	2 cap/3L	4G/5G	5Y/6R
1J/2J	3A/3+	$\begin{pmatrix}4H/\Omega_{22}\\ \Omega_{22}/4V\end{pmatrix}$	5 cap/6L
1R/1W	3B/3C	4J/5J	6A/6+
1S/1−	3C/3H	4R/4W	6B/6C
1T/1U	3E/3−	4S/4X	6C/6H
1X/2V	$\begin{pmatrix}3F/\Omega_{B3}\\ \Omega_{B3}/3+\end{pmatrix}$	4T/4U	6E/6Y
1Y/2N	3F/3S	4T/5K	$\begin{pmatrix}6F/\Omega_{B6}\\ \Omega_{B6}/6+\end{pmatrix}$
1 cap/2L	3G/4G	4W/5N	6F/6T
2A/2+	$\begin{pmatrix}3H/\Omega_{22}\\ \Omega_{22}/3V\end{pmatrix}$	4Y/5V	$\begin{pmatrix}6H/\Omega_{22}\\ \Omega_{22}/6R\end{pmatrix}$
2B/2C	3J/4J	4 cap/5L	6S/6 cap
2C/2H	3R/3W	5A/5+	6T/6X
2E/2−	3S/3X	5B/5C	6Z/6−
$\begin{pmatrix}2F/\Omega_{B2}\\ \Omega_{B2}/2+\end{pmatrix}$	3T/3U	5C/5H	
		5E/5−	

Output lights 1 through 6 are used to display the computer's results in binary:

Light	1	2	3	4	5	6
Weight	32	16	8	4	2	1

(The limit of this accumulator is 111111 or 63.)

Slide switches 2 through 6 are used for input. (Normal position is LEFT.)

Slide switch 1 RIGHT is the ***shift*** position.

Slide switch 1 LEFT is the ***add*** position.

The accumulator is re-set by turning power off, then on again.

Procedure:

The general procedure for any multiplication is:

a) Multiply the multiplicand by the low order (right-most) bit of the multiplier.

b) Add the result of (a) into the accumulator.

c) Shift left one bit.

Note: There is no shift after the product of the multiplicand and the high order (left-most) bit of the multiplier has been added.

These steps are repeated, using each bit of the multiplier in turn.

To add a number into the accumulator:

All slide switches must be LEFT. Enter a binary number using slide switches 2 through 6: To enter "1," move the slide switch RIGHT, then back to LEFT. To enter "0," do not move the slide switch.

The number entered is displayed on the output lights. To shift left one bit:

Set slide switch 1 RIGHT. Push pushbuttons 1 through 6 in sequence. Set slide switch 1 LEFT. The shifted number is displayed on the output lights.

As an example, we will multiply:

5×11 or 101×1011

multiplicand $= 1011$ (11)

multiplier $= 101$ (5)

1. Set all slide switches LEFT and turn power on. Multiply the multiplicand by the low order bit of the multiplier:

   $1 \times 1011 = 1011$

   Enter the product (1011) by moving slide switches 3, 5 and 6 RIGHT, then back LEFT.

2. Shift left one bit:

   Set slide switch 1 RIGHT; push pushbuttons 1 through 6 in sequence. Set slide switch 1 LEFT. The output lights now show:

   010110

3. Multiply the multiplicand by the next higher bit of the multiplier:

   $0 \times 1011 = 0000$

   This is "entered" by ***not*** moving any of the slide switches.

4. Shift left one bit (as in step 2).

5. Multiply the multiplicand by the high order bit of the multiplier:

   $1 \times 1011 = 1011$

   Enter the product (1011) by moving slide switches 3, 5 and 6 RIGHT, then back LEFT.

6. The output lights are now displaying the final product:

   110111 or 55

The operations carried out by the computer were:

```
  1011
   101
------
  1011
 0000
1011
------
110111
```

This program may be left wired for the experiment which follows.

Experiment 15: Division with an Accumulator

In Book IV we explained how a full-scale computer divides by subtracting and shifting (see Book IV, page 82 ff). An accumulator which was capable of shifting and subtracting was used to carry out the required steps in a division.

The accumulator from Experiment 14 can be modified to perform divisions. If you have the accumulator from the previous experiment wired on the MINIVAC, convert it to an accumulator for division by making the following changes:

Remove

1A/1V
1F/2N
1G/1R
(1H/Ω_{22} / Ω_{22}/2K)
1J/1U
1S/1−
1Y/2N
5Y/6R
6E/6Y
6F/6T
(6H/Ω_{22} / Ω_{22}/6R)
6S/6 cap
6T/6X
6Z/6−

Add

1F/1S
(1H/Ω_{22} / Ω_{22}/1V)
1S/1Y
1T/2K
1W/2N
5Y/6H
6A/6S
6E/6−
6F/6Y
6G/6T
6J/6W
6R/6W
6S/6X
6T/6U
6V/6−

The complete program for an accumulator for division is:

1A/1+
1B/1C
1C/1H
1E/1−
(1F/Ω_{B1} / Ω_{B1}/1+)
1F/1S
1G/2G
(1H/Ω_{22} / Ω_{22}/1V)
1J/2J
1R/1W
1S/1Y
1T/1U
1T/2K
1W/2N
1X/2V
1 cap/2L
2A/2+
2B/2C
2C/2H
2E/2−
(2F/Ω_{B2} / Ω_{B2}/2+)
2F/2S
2G/3G
(2H/Ω_{22} / Ω_{2B}/2V)
2J/3J
2R/2W
2S/2X
2T/2U
2T/3K
2W/3N
2Y/3V
2 cap/3L
3A/3+
3B/3C
3C/3H
3E/3−
(3F/Ω_{B3} / Ω_{B3}/3+)
3F/3S
3G/4G
(3H/Ω_{22} / Ω_{22}/3V)
3J/4J
3R/3W
3S/3X
3T/3U
3T/4K
3W/4N
3Y/4V
3 cap/4L
4A/4+
4B/4C
4C/4H
4E/4−
(4F/Ω_{B4} / Ω_{B4}/4+)
4F/4S
4G/5G
(4H/Ω_{22} / Ω_{22}/4V)
4J/5J
4R/4W
4S/4X
4T/4U
4T/5K
4W/5N
4Y/5V
4 cap/5L
5A/5+
5B/5C
5C/5H
5E/5−
(5F/Ω_{B5} / Ω_{B5}/5+)
5F/5S
5G/6G
(5H/Ω_{22} / Ω_{22}/5V)
5J/6J
5R/5W
5S/5X
5T/5U
5T/6K
5W/6N
5Y/6H
5 cap/6L
6A/6+
6A/6S
6B/6C
6C/6H
6E/6−
(6F/Ω_{B6} / Ω_{B6}/6+)
6F/6Y
6G/6T
6J/6W
6R/6W
6S/6X
6T/6U
6V/6−

Relay indicator lights 1 through 6 will first display the dividend; as the division proceeds, these lights will display the partial (remaining) dividend. The final result displayed on the relay indicator lights to the ***left*** of the binary point is the ***remainder.*** The quotient must be recorded by the operator; it will not be displayed by the machine.

Slide switches 1 through 5 are used to subtract the divisor from the dividend. The normal position for these slide switches is LEFT.

Slide switch 6 RIGHT is the ***shift*** position.

Slide switch 6 LEFT is the ***subtract*** position.

Procedure:

The general rules for any division are:

The dividend is limited to five significant bits. (This means that originally there will always be a zero in the high order stage of the six-bit accumulator.

There will always be a zero to the left of the divisor's most significant bit. (For example, a divisor of 110 will be written as 0110.)

The general procedure is:

a. Attempt to subtract the divisor from the high order bits of the dividend. If the subtraction can be carried out, enter "1" in the "quotient record." If the subtraction can ***not*** be carried out, enter "0" in the quotient record. If possible, perform the subtraction on the machine.

b. Shift left one bit. (Note: there will always be as many shifts as the difference between the length of the dividend (six bits) and the length of the divisor.

When the division begins, the binary point is to the right of the low order bit of the dividend (that is, to the right of section 6). When a shift is carried out, the binary point also shifts. When the division has been completed, the binary point will be displaced to the left by the number of shifts which were carried out.

To subtract a number from the display in the relay indicator lights:

All slide switches should be LEFT.
To subtract "1," move the slide switch RIGHT, then LEFT again.
To subtract "0," do not move the slide switch.

To shift left one bit:

Set slide switch 6 RIGHT. Push pushbuttons 1 through 6 in sequence. Set slide switch 6 LEFT.

As an example we will divide:

$11010 \div 11$ or $26 \div 3$

dividend = 11010 (26)
divisor = 11 (3)

The dividend must always be six bits long. We will therefore add a zero to the left of the most significant bit:

$011010 = 26$

We must also place a zero to the left of the most significant bit of the divisor:

$011 = 3$

Since there are now six bits in the dividend and three bits in the divisor, there will be $6 - 3 = 3$ shifts in this division.

1. Set all slide switches LEFT and turn power on. Enter the dividend (011010) on the relay indicator lights:

   Starting with section 6, turn off the appropriate relays by touching positive power to the "F" terminal. In this case, we will touch a wire from M+ to 6F, 4F and 1F, in that order.

   On a piece of paper, make a "quotient record" like this:

PARTIAL DIVIDENDS	SUBTRACTION?	QUOTIENT

2. Enter the dividend which is now being displayed on the relay indicator lights in the "PARTIAL DIVIDENDS" column; enter the divisor under the high order bits of this dividend.

   We will now inspect the high order bits of the dividend above the advisor:

   Can the divisor be subtracted from this portion of the dividend? The answer is YES; we enter "1" in the quotient column.

   The quotient record now looks like this:

PARTIAL DIVIDENDS	SUBTRACTION?	QUOTIENT
011010 011	yes	1

Carry out the subtraction by setting slide switches 2 and 3 RIGHT, then LEFT again.

3. Shift left one bit by setting slide switch 6 RIGHT, pushing pushbuttons 1 through 6 in order and returning slide switch 6 LEFT. The partial dividend is now being displayed on the relay indicator lights. We will indicate the shift on our quotient record by an arrow; we will enter the partial dividend beneath the arrow and will again enter the divisor:

PARTIAL DIVIDENDS	SUBTRACTION?	QUOTIENT
011010 011	yes	1
← 000100 011		

4. Again we inspect the high order bits of the dividend above the divisor:
   Can the divisor be subtracted from this portion of the dividend? The answer is NO; we enter "0" in the "QUOTIENT" column.
5. Shift left one bit (indicate this on the quotient record by an arrow). Enter the new partial dividend with the divisor beneath it as before:

PARTIAL DIVIDENDS	SUBTRACTION?	QUOTIENT
011010 011	yes	1
← 000100 011	no	0
← 001000 011		

6. We inspect the high order bits of the dividend above the divisor:
   Can the divisor be subtracted from this portion of the dividend? The answer is NO; we enter "0" in the "QUOTIENT" column.
7. Shift left one bit and enter the new partial dividend in the quotient record with the divisor beneath it. A subtraction cannot be carried out, so the quotient record now looks like this:

PARTIAL DIVIDENDS	SUBTRACTION?	QUOTIENT
011010 011	yes	1
← 000100 011	no	0
← 001000 011	no	0
← 010000 011	no	0

8. We have now carried out three shifts; the division is finished. Reading down the "QUOTIENT" column on our quotient record, the quotient is:

$$1000 = 8$$

The binary point has shifted three places to the left, so the remainder (from the relay indicator lights) is:

$$010 \wedge 000 = 2$$

Thus:

$$011010 \div 011 = 1000, \text{ remainder } 010$$

or

$$26 \div 3 = 8, \text{ remainder } 2$$

Experiment 16: Ring-Tail Switch-Tail Shift Register

"Ring-tail switch-tail" is a phrase used to describe a closed, complementing shift register — that is, a shift register whose end-around carry is complemented. For example, if a zero is held in the high order stage of this shift register, when the shift command is given the zero of the high order stage will be complemented so that a one is entered into the low order stage of the shift register.

The shift register is, of course, a combination of flip-flops. When flip-flops are combined to form a ring-tail switch-tail shift register, the unit can be wired so that a change in a **specific** flip-flop will produce an output pulse. Thus, if the register is made up of N stages (N flip-flops), the frequency of output pulses will be 1/2N times the frequency of the input pulses.

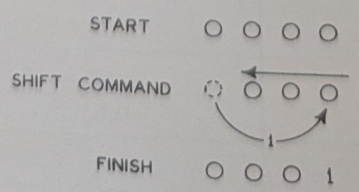
When a ring-tail switch-tail shift register is wired like this, it is a ***scaler*** — it scales input pulses down by a specific factor. This characteristic makes the ring-tail switch-tail shift register particularly valuable in the design and construction of high-speed computers. It permits the use of slower (and therefore cheaper) flip-flops in counters which must accept high speed input.

The program for a ring-tail switch-tail shift register is:

1E/1−	2E/2−	3J/3−	5F/6G
1F/6F	$\begin{pmatrix}2F/\Omega_{B2}\\ \Omega_{B2}/2+\end{pmatrix}$	4C/4H	$\begin{pmatrix}5F/\Omega_{B5}\\ \Omega_{B5}/5+\end{pmatrix}$
1F/1Y		4E/4−	
1G/2F	2G/2−	4F/6K	5G/5+
1H/1 cap	2H/6 com	$\begin{pmatrix}4F/\Omega_{B4}\\ \Omega_{B4}/4+\end{pmatrix}$	$\begin{pmatrix}5H/\Omega_{22}\\ \Omega_{22}/6N\end{pmatrix}$
$\begin{pmatrix}1J/\Omega_{22}\\ \Omega_{22}/3H\end{pmatrix}$	2J/2+		
	2K/1+	4G/4+	5J/5−
1K/3F	3C/3H	$\begin{pmatrix}4H/\Omega_{22}\\ \Omega_{22}/4\text{ com}\end{pmatrix}$	5 cap/6L
1L/2 cap	3E/3−		6E/6−
1N/4 com	$\begin{pmatrix}3F/\Omega_{B3}\\ \Omega_{B3}/3+\end{pmatrix}$	4J/4−	6H/6 cap
1X/1+		5C/5H	$\begin{pmatrix}6J/\Omega_{22}\\ \Omega_{22}/6\text{ com}\end{pmatrix}$
2C/2L	3G/3+	5E/5−	

Procedure:

1. Turn power on. The shift register is storing 0000 (relay indicator lights 2 through 5 are off).
2. Push pushbutton 1. The zero in the high order stage of the shift register is complemented and shifted into the low order stage:



3. Push pushbutton 1. Once again, the high order stage is complemented and shifted into the low order stage. The shift register is now storing 0011.

4. Push pushbutton 1 twice. The shift register has now completed one half of its full cycle and is storing 1111.

5. Continue pushing pushbutton 1; the shift register will complete its eight-step cycle and return to 0000.

Note: You can set the shift register so that its original information is any number from 0 through 15 by closing the appropriate relays. For example, to set the shift register for 1010:

Turn power on. Plug one end of a programming wire into M+. Touch the free end of this wire to 2F and 4F. The register is now displaying 1010.

To make the shift register operate automatically, add these connections:

1Y/D1	D17/M+
D16/M+	D18/M−

Each time the rotary switch passes 1, the register will shift.

Experiment 17: Electronic Memory

This program uses five of the capacitors as internal storage elements. A charged capacitor represents "one"; a discharged capacitor represents "zero." The relays are used to "read out" the information stored in the capacitors. When the information is displayed, the relays are also acting as external storage elements.

Information is entered into this memory unit in binary form, using five pushbuttons. In full-scale systems, information is usually entered into the central processing unit which then directs the information to the appropriate memory unit. Some new systems use a special-purpose subsidiary computer as an input-output unit to the memory unit. With such a subsidiary system, the central processing unit is directed to the memory unit for required information. This method is particularly valuable when input-output time exceeds actual program execution time. Using the subsidiary unit for input-output frees the main computer for program execution and can thus increase the speed of the computer.

The program for an electronic memory unit is:

1A/1H	2G/3G	4F/D14	(5 com/Ω_{22} Ω_{22}/6 com)*
1B/1+	2H/2X	4G/5G	6A/6K
1B/2B	2J/3J	4H/4X	6B/6+
1E/2E	2Y/2 cap	4J/5J	6E/6H
(1F/Ω_{B1} Ω_{B1}/2+)	2 cap/D6	4Y/4 cap	6F/6−
1F/D8	3A/3H	4 cap/D15	6G/6 com
1G/1−	3B/4B	5A/5H	6G/6 cap
1G/2G	3E/4E	5E/6W	6H/6S
1H/1X	(3F/Ω_{B3} Ω_{B3}/3+)	(5F/Ω_{B5} Ω_{B5}/5+)	6L/6W
(1J/Ω_{22} Ω_{22}/1+)	3F/D2	5F/D11	6L/D17
1J/2J	3G/4G	5H/5X	6T/D16
1Y/1 cap	3H/3X	5U/5−	6V/6−
1 cap/D9	3J/4J	5V/D16	D1/M−
2A/2H	3Y/3 cap	5Y/5 cap	D1/D4
2B/3B	3 cap/D3	5 cap/D12	D4/D7
2E/3E	4A/4H	5 com/6+	D7/D10
(2F/Ω_{B2} Ω_{B2}/2+)	4B/5B	(5 com/Ω_{22} Ω_{22}/6 com)*	D10/D13
2F/D5	4E/5E	(5 com/Ω_{22} Ω_{22}/6 com)*	D18/M+
	(4F/Ω_{B4} Ω_{B4}/4+)		

* There are *three* 22 ohm resistors between 5 com and 6 com.

Note: The operation of this circuit depends entirely upon the pull-in characteristics of relay 6. If the circuit does not operate adequately — specifically, if relays 1 through 5 drop out as the rotary switch turns when the unit is "reading out" — you should transfer the connections on relay 6 to another relay. Select that relay which pulls in most easily — the relay which will pull in with the highest resistance in series with the relay coil. Then exchange connections between that relay and relay 6, with one exception: preserve the connection to C on the better relay. That is, you will have the connection 6G/?C rather than 6G/6C.

Procedure:

1. Set slide switch 5 RIGHT; set slide switch 6 LEFT; set the rotary switch to 0. Turn power on.
2. Clear memory unit:
   Set slide switch 5 LEFT. Set slide switch 6 RIGHT and let the rotary switch make one complete revolution. Set slide switch 6 LEFT to stop the rotary switch. Be sure that the rotary switch is stopped at 0.
   Set slide switch 5 RIGHT.
3. Enter Data:
   Using pushbuttons 1 through 5, enter data in binary. To enter "zero," do not push a pushbutton. To enter "one," push a pushbutton and hold it down long enough for the capacitor to charge. For example, to enter 10101, push pushbuttons 1, 3 and 5.
   The capacitors are now storing the information entered into the memory unit. A charged capacitor represents "one"; a discharged capacitor represents "zero."
4. Read out contents of memory unit:
   Set slide switch 6 RIGHT. At the rotary switch turns, the information in the capacitors is transferred to the relays; the information is now displayed on output lights 1 through 6.
5. To enter new data while the memory unit is displaying its contents, push pushbuttons 1 through 5. Pushing a pushbutton at this stage ***complements*** the information in the memory unit. For example, if 10101 is being displayed:
   to enter the new data 10011, push pushbuttons 3 and 4.

When pushing a pushbutton to enter new data, do not hold the pushbutton down.

Experiment 18: ESP Tester

The electronic memory circuit can be modified to build an ESP (extra-sensory perception) tester. You can enter any one of 31 light combinations and challenge your friends to attempt to match the combination you have entered. Their success will depend upon their ability to figure out — either by "reading your mind" or by guessing — what you have entered.

The program for an ESP tester is:

1A/2A
1B/1W
1C/1G
1E/1−
1F/D2
$\begin{pmatrix} 1F/\Omega_{B1} \\ \Omega_{B1}/1+ \end{pmatrix}$
1H/2H
1K/1T
1L/6B
1N/1R
1S/2S
1V/2V
1X/1 com
1X/2X
1Y/1 cap
$\begin{pmatrix} 1\ com/\Omega_{22} \\ \Omega_{22}/1+ \end{pmatrix}$
1 cap/D1
2A/3A
2B/2W
2C/2G
2E/2−
2F/D5
$\begin{pmatrix} 2F/\Omega_{B2} \\ \Omega_{B2}/2+ \end{pmatrix}$
2H/3H
2K/2T
2L/3L
2N/2R
2V/3V
2X/3X
2Y/2 cap
2 cap/D4
3A/4A
3B/3W
3C/3G
3E/3−
3F/D8
$\begin{pmatrix} 3F/\Omega_{B3} \\ \Omega_{B3}/3+ \end{pmatrix}$
3H/4H
3K/3T
3N/3R
3S/4S
3V/4V
3X/4X
3Y/3 cap
3 cap/D7
4A/5A
4B/4W
4C/4G
4E/4−
4F/D11
$\begin{pmatrix} 4F/\Omega_{B4} \\ \Omega_{B4}/4+ \end{pmatrix}$
4H/5H
4K/4T
4L/5L
4N/4R
4V/5V
4X/5X
4Y/4 cap
4 cap/D10
5A/4+
5B/5W
5C/5G
5E/5−
5F/D14
$\begin{pmatrix} 5F/\Omega_{B5} \\ \Omega_{B5}/5+ \end{pmatrix}$
5H/5+
5K/5T
5N/5R
5S/6U
5V/6U
5Y/5 cap
5 cap/D13
6A/6+
6E/6−*
6F/6L*
$\begin{pmatrix} 6K/\Omega_{22} \\ \Omega_{22}/6+ \end{pmatrix}$*
6L/6T*
6R/D19
6S/D16
6T/6Y
6V/6X
6V/D18
6W/D12
D3/D6
D3/D15
D6/D9
D9/D12
D15/M−
D17/M+
D18/M−

* As in the previous experiment, the operation of this circuit depends entirely upon the operation of relay 6. If this circuit does not operate correctly, transfer the connections to relay 6 to another relay. Select that relay which pulls in most easily — that is, the relay which will pull in with the highest resistance in series with the relay coil. Then exchange connections between that relay and relay 6, with one exception: preserve the connection to C on the better relay. For example, if relay 4 pulls in more easily than relay 6, exchange the connections on relay 4 with the connections on relay 6. (Exchange ***only*** the ***relay*** connections — C, E, F, G, H, J, K, L, N). However, do not remove the connection to 4C. You will thus have the connection 4C/6G.

Procedure:

1. Set all slide switches LEFT and turn power on.

2. Clear the memory:

   Set slide switch 6 RIGHT; hold pushbutton 6 down while the rotary switch makes a complete revolution. Light 6 will come on indicating that the display on relay indicator lights 1 through 5 is the same as the display on binary output lights 1 through 5. Set slide switch 6 LEFT when the memory has been cleared.

3. You are now ready to experiment. Without letting the subject see what you are entering, enter a light combination on pushbuttons 1 through 5. (Push a pushbutton to indicate a light on.) Since the capacitors are storing the information you have entered, there will be no visible indication of your light combination.

4. Ask the subject to attempt to match your combination. The subject can turn on an output light by moving a slide switch RIGHT. The subject may move any of slide switches 1 through 5. Output lights 1 through 5 will display the subject's combination.

5. Compare the experimenter's combination with the subject's combination:

   Set slide switch 6 RIGHT and allow the rotary switch to make one full revolution. Set slide switch 6 LEFT. The experimenter's combination now appears on relay indicator lights 1 through 5; the subject's combination appears on the output lights. If the combinations are the same, output light 6 will come on.

Set all slide switches LEFT, then clear the memory to try the experiment again.

Note: Remember that the bias characteristics of a relay may change when the relay is wired into a circuit. If you are having difficulty with a program, try using different bias resistances for the relays to improve the operation of the circuit.

Experiment 19: Tic-Tac-Toe

Version I

The addition of capacitors and resistors allows us to build a much more sophisticated circuit for tic-tac-toe. This improved circuit is simpler in operation than the previous tic-tac-toe circuit (see Book VI, page 53). As with the previous circuit, MINIVAC cannot lose.

The program for tic-tac-toe is:

1 cap/1+
1 cap/2 cap
1R/M10
1S/D9
1T/1−
1T/1W
1V/6B
2E/2−
$\begin{pmatrix} 2F/\Omega_{B2} \\ \Omega_{B2}/2+ \end{pmatrix}$
2F/6J
2G/2−
$\begin{pmatrix} 2H/\Omega_{33} \\ \Omega_{33}/2\ com \end{pmatrix}$
2J/2+
2L/3K
2N/6K
2R/2W
2S/4K
2T/2U
2T/5R
2V/4N
2W/5T
2 com/6G
3C/3−
3F/M11
3G/5V
3H/6X
3J/6A
3K/5N
3L/5L
3N/4N
3 com/M5b
3 com/D5
3 com/M1t
3 com/M5t
4E/4—
$\begin{pmatrix} 4F/\Omega_{B4} \\ \Omega_{B4}/4+ \end{pmatrix}$
4F/5J
4G/4−
4G/4L
$\begin{pmatrix} 4H/\Omega_{22} \\ \Omega_{22}/5G \end{pmatrix}$
4J/4+
4 com/M6b
4 com/D6
4 com/M2t
4 com/M6t
4 cap/5 cap
5A/5U
5B/5−
5E/5+
5F/6F
5H/5 cap
5K/5S
5 com/M7b
5 com/D7
5 com/M3t
5 com/M7t
6B/D0
6E/6−
6F/D16
6H/6 cap
6L/D19
6X/M10
6Y/M+
6 com/M8b
6 com/D8
6 com/M4t
6 com/M8t
D1/D5
D1/M 1b
D2/D6
D2/M2b
D3/D7
D3/M3b
D4/D8
D4/M4b
D17/M10
D18/M−
$\begin{pmatrix} D18/\Omega_{22} \\ \Omega_{22}/D19 \end{pmatrix}$
M9b/M11
M10/free*
M11/free**

* This is the machine's "play" wire.
** This is the player's "play" wire.

The general rules for playing tic-tac-toe with this program are:

The machine's "play" wire is always placed in a BOTTOM terminal on the game matrix.

The player's "play" wire is always placed in a TOP terminal on the game matrix.

All slide switches should be RIGHT ***except*** the slide switch corresponding to the move. (For example, on the third move, slide switch 3 is LEFT; all other slide switches are RIGHT.)

Power must be turned off when moves are being indicated on the game matrix.

Light 5 on indicates a tie game.

Light 6 on indicates that the machine has won.

The machine must always make the first move.

Note: It is most convenient to keep track of the game on a piece of paper.

To play:

1. Set all slide switches RIGHT. Be sure that the machine's "play" wire and the player's "play" wire are both free.

2. The first move:

   Set slide switch 1 LEFT. Turn power on. Push pushbutton 6 and hold it down until the rotary switch stops. The rotary switch is now indicating the machine's first move. Turn power off. Set slide switch 1 RIGHT.

   Indicate the machine's first move on the game matrix by plugging the machine's "play" wire into a BOTTOM terminal corresponding to the number indicated on the rotary switch. The player indicates his first move by plugging play wire into a TOP terminal of the square he wants to play. Set the rotary switch at the ***player's*** first move.

3. The second move:
   Set slide switch 2 LEFT and turn power on. Push pushbutton 6 and hold it down until the rotary switch stops. The rotary switch is now indicating the machine's second move. Turn power OFF. Set slide switch 2 RIGHT. Indicate the machine's second move on the game matrix as before. The player indicates his second move on the game matrix. Be sure that the rotary switch is pointing ***directly*** at the ***machine's*** second move.

4. All succeeding moves are carried out as in step 3 above, using the slide switch corresponding to the number of the move. After the machine's last move, either light 5 or light 6 will come on indicating the results of the game.

(Note: Always be sure that relays 5 and 6 are both ON before any move is made.)

Version II

The following program is provided for the benefit of those people who become discouraged at losing to a machine every time. With this program, the computer will sometimes lose. It will also try to cheat and will require disciplining to force it to behave.

The "stubborn program" for tic-tac-toe is:

1A/1J	2E/Ω_{22}	3H/3+	5G/Ω_{150}
1B/1C	Ω_{22}/2 cap	3K/D9	Ω_{150}/5 cap
1C/1−	2F/2G	3S/M4t	5H/5+
1E/Ω_{22}	2F/2 com	3V/M5t	5N/D8
Ω_{22}/1 cap	2G/Ω_{150}	3 com/M6t	5 com/M8t
1F/1G	Ω_{150}/2 cap	3 com/M6b	5 com/M9b
1F/1 com	2H/2+	4A/4J	6A/6J
1G/Ω_{150}	2H/2L	4B/4C	6B/6C
Ω_{150}/1 cap	2K/3N	4C/4−	6C/6−
1H/1+	2K/4 com	4E/Ω_{22}	6E/Ω_{22}
1K/5L	2R/6K	Ω_{22}/4 cap	Ω_{22}/6 cap
1L/1R	2S/4V	4F/4G	6F/6G
1L/3N	2T/4L	4F/4 com	6F/6 com
1N/1V	2W/D3	4G/Ω_{150}	6G/Ω_{150}
1N/2V	2X/D9	Ω_{150}/4 cap	Ω_{150}/6 cap
1R/3X	2Z/D5	4H/4+	6H/6+
1S/1−	2Y/4T	4K/4S	6L/D6
1S/2S	2Z/3Y	4K/5K	6X/6+
1T/3L	2 com/5 com	4L/5L	6Y/D17
1W/D2	2 com/3R	4N/4U	6 com/M9t
1X/D4	3A/3J	4 com/M7b	D16/D19
1Y/4N	3B/3C	4 com/M7t	D18/M−
1Z/D7	3C/3−	5A/5J	M10/M+
1 com/M2t	3E/Ω_{22}	5B/5C	M10/M11
1 com/M3t	Ω_{22}/3 cap	5C/5−	M10/free*
2A/2J	3F/3G	5E/Ω_{22}	M11/free**
2B/2C	3F/3 com	Ω_{22}/5 cap	
2C/2−	3G/Ω_{150}	5F/5G	
	Ω_{150}/3 cap	5F/5 com	

* This is the machine's "play" wire.
** This is the player's "play" wire.

The general rules for play are:

The machine's "play" wire is always placed in a BOTTOM terminal on the game matrix.
The player's "play" wire is always placed in a TOP terminal on the game matrix.
All slide switches should be right ***except*** the slide switch corresponding to the move.
The machine always moves first; its first move is to 1 on the game matrix and must be indicated by the player.
When indicating a move on the game matrix, do ***not*** let the play wires touch any matrix terminal other than the proper one.
The game is re-set by turning the power switch off, then on again.

To play:

1. Set all slide switches RIGHT. Turn power on.
2. The machine moves first. This first move is to 1 on the game matrix; indicate the machine's first move by:
   setting the rotary switch dial to 1; placing the free end of the machine's "play" wire in a BOTTOM terminal on 1 of the game matrix.
3. The player indicates his first move by placing the free end of his "play" wire in a TOP terminal on the game matrix.
4. The first move is now completed. Set slide switch 1 LEFT. Hold pushbutton 6 down until the rotary switch stops. The rotary switch stops at the machine's next move.
5. Set slide switch 1 RIGHT and indicate the machine's last move (from the rotary switch) by moving the free end of the machine's "play" wire to the corresponding BOTTOM terminal on the game matrix.
6. The player indicates his next move by moving the free end of his "play" wire to a TOP terminal on the game matrix.
7. The second move is now completed. Set slide switch 2 LEFT. Hold pushbutton 6 down until the rotary switch stops. The rotary switch stops at the machine's next move.

Steps 5 through 7 are repeated, using slide switch 3 for the third move, 4 for the fourth move, etc. The machine will not indicate the results of the game; the player must keep track of the game so that he will know when he has won.

Occasionally the machine will cheat by trying to move into an occupied square; or, it may stubbornly refuse to move at all. To discipline the machine, hold down pushbuttons 1 and 2 while pushing pushbutton 6. If the machine still insists upon cheating, release all three pushbuttons and push pushbuttons 3 and 6.

Experiment 20: Music Composer I

There are a number of programs in existence which allow full-scale computers to "compose" music. Many of these programs are based on one of two fundamental techniques:

The "functional" technique: Specific rules for harmony, rhythm, sequence of notes, etc. are programmed as mathematical functions. The computer is then programmed to generate random numbers; each random number is substituted for a variable in the functions. Every random number generated eventually becomes a note.

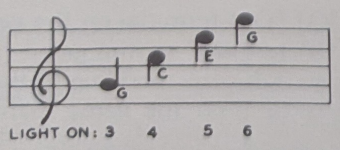
The "sampling" technique: Specific rules for harmony, rhythm, sequence of notes, etc. are programmed as mathematical functions. The computer is then programmed to randomly select notes; each note is compared against the rules. If a randomly generated note does not agree with the rules, it is discarded and another note is selected.

We can program the MINIVAC to "compose" music using the "functional" technique. The rules must, of course, be very simple. The rules which we will use are:

There are only four possible notes.

There are only two possible time values for each note.

When the program is wired, the four possible notes will be indicated by output lights 3 through 6. Light 2 will indicate the time value of each note.



LIGHT 2 OFF = ♩ (QUARTER NOTE)
LIGHT 2 ON = ♪ (EIGHTH NOTE)

The program for this music composer is:

1A/1−	3B/4B	$\begin{pmatrix}4F/\Omega_{B4}\\ \Omega_{B4}/4+\end{pmatrix}$	5G/6J
1B/2F	3E/3−	4G/5G	5H/6F
1X/1+	$\begin{pmatrix}3F/\Omega_{B3}\\ \Omega_{B3}/3+\end{pmatrix}$	4J/5J	5J/6G
1Y/2F	3F/4H	4L/6X	5L/5+
2A/2+	3G/4J	5A/5K	5N/6A
2B/3L	3H/4F	5B/6B	6B/6N
2E/2−	3J/4G	5E/5−	6E/6−
2G/2−	3N/4K	5F/6H	$\begin{pmatrix}6F/\Omega_{B6}\\ \Omega_{B6}/6+\end{pmatrix}$
2H/3G	4A/5A	$\begin{pmatrix}5F/\Omega_{B5}\\ \Omega_{B5}/5+\end{pmatrix}$	6L/6X
2K/2+	4B/6K		6Y/6−
2L/3J	4E/4−		
3A/6A			

Pushbutton 1 is the "create" button.
Pushbutton 6 is the "compose" button.

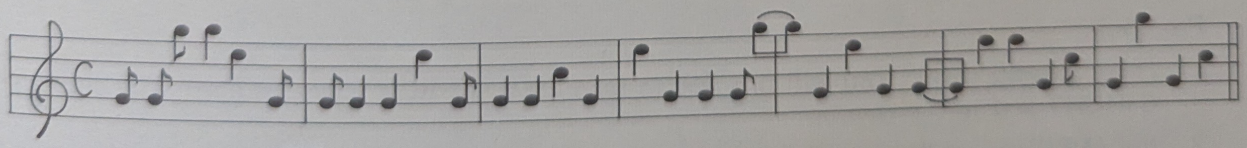
Procedure:

1. Draw a treble staff. You will use this to record MINIVAC's "composition." Turn power on.
2. Push the "create" button (pushbutton 1) and hold it down. When you feel that MINIVAC has had sufficient time to select its note, release the "create" pushbutton.
3. Push the "compose" button (pushbutton 6). When you hold the "compose" button down, MINIVAC will display its selection on output lights 2 through 6.

   Light 2 OFF = quarter note
   Light 2 ON = eighth note
   Light 3 ON = G
   Light 4 ON = C
   Light 5 ON = E
   Light 6 ON = G

   Record this note on your staff.
4. Repeat steps 2 and 3, recording each note as it is displayed. You will determine the length of the "composition" by the number of times you repeat steps 2 and 3.
5. Now select a time signature for the piece. (You will probably want to use either $\frac{3}{4}$ or $\frac{4}{4}$). Mark the measures of the composition accordingly.

   This is a sample of a MINIVAC composition:



(Note that at the end of the third and fourth measures "holds" are indicated. These are required when MINIVAC has selected an odd number of eighth notes for a measure and the final note in the measure is a quarter note.)

Note: Be sure that the bias resistors are correct for the circuit. An incorrect bias resistance will cause the machine to favor one note.

If you want to change the frequency of appearance of a particular note, change the corresponding relay's bias resistance.

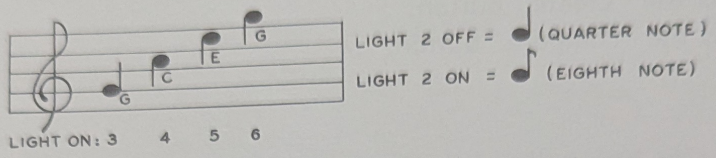
Experiment 21: Music Composer II

The preceding "composition" circuit used only two rules. By adding a third rule, we will be able to write more music. The three rules which will be programmed are:

There are only four possible notes.
There are only two possible time values for each note.
Eighth notes must always appear in pairs.

The addition of the third rule means that we will not have quarter notes held across a measure. Pieces written with this program make fire bugle calls when put into $\frac{4}{4}$ time.

The output lights will be used to indicate notes as in the previous experiment:



The program for the "three-rule" music composer is:

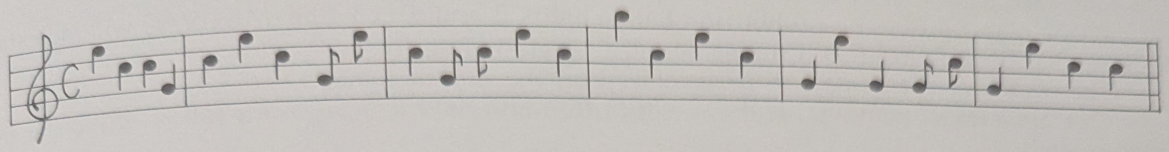
$\begin{pmatrix}1E/\Omega_{B1}\\ \Omega_{B1}/1+\end{pmatrix}$	1V/2+	3J/4G	5B/6B
1E/1J	1W/2F	3K/4N	5E/5−
1F/1−	1W/6L	3N/4L	$\begin{pmatrix}5F/\Omega_{B5}\\ \Omega_{B5}/5+\end{pmatrix}$
$\begin{pmatrix}1H/\Omega_{22}\\ \Omega_{22}/2K\end{pmatrix}$	2A/2F	4A/5A	5F/6H
1H/3L	2E/2−	4B/6K	5G/6J
1J/2N	2G/4F	$\begin{pmatrix}4E/\Omega_{B4}\\ \Omega_{B4}/4+\end{pmatrix}$	5H/6F
1K/2B	2L/2 cap	4F/4 cap	5J/6G
1L/1−	3A/6A	4G/5G	5L/5−
1L/2H	3B/4B	4J/5J	5N/6A
1N/2G	3E/3−	4K/6L	6B/6N
1R/3G	$\begin{pmatrix}3F/\Omega_{B3}\\ \Omega_{B3}/3+\end{pmatrix}$	4N/4−	6E/6−
1S/2−	3F/4H	$\begin{pmatrix}4\text{ cap}/\Omega_{120}\\ \Omega_{120}/5+\end{pmatrix}$	$\begin{pmatrix}6F/\Omega_{B6}\\ \Omega_{B6}/6+\end{pmatrix}$
1U/3J	3G/4J	5A/5K	
	3H/4E		

Slide switch 1 LEFT = "create"
Slide switch 1 RIGHT = "compose"

Procedure:

1. Draw a treble staff. You will use this to record MINIVAC's "composition". Set slide switch 6 RIGHT and turn power on.
2. Put MINIVAC in the "create" mode (set slide switch 6 LEFT). When you feel that MINIVAC has had sufficient time to select its note, put the machine in the "compose" mode (set slide switch 6 RIGHT).
3. MINIVAC is now displaying its selection on output lights 2 through 6 (see preceding convention). Record this note on your staff.
4. Repeat steps 2 and 3, recording each note as it is displayed. You will determine the length of the "composition" by the number of times you repeat steps 2 and 3.
5. Now select a time signature for the piece and mark the measures.

This tune was composed on the MINIVAC using the three rule program. Note that eighth notes always appear in pairs.



(As with the previous circuit, the frequency of appearance of a particular note can be changed by changing the corresponding relay's bias resistor.)

Experiment 22: Team Reaction Tester

This program is an expanded version of the "Reaction Time Tester" in Book VI (see page 60) which allows two teams of two persons each to compete. Each team of two persons tries to push its pushbuttons down ***first*** when the signal is given. However, this game has the additional requirement of coordination for a team to win: a team's two pushbuttons must be pushed at ***exactly*** the same time for a win.

Team A uses pushbuttons 1 and 2.
Team B uses pushbuttons 5 and 6.
Pushbutton 3 is the re-set button.
Light 1 on indicates that Team A wins.
Light 2 on indicates that Team A has fouled.
Light 5 on indicates that Team B has fouled.
Light 6 on indicates that Team B wins.
Lights 3 and 4 on give the signal "START".
Lights 1 and 2 ***both*** on indicate:

That Team A was first, but the players were not coordinated. Or, after Team A has won, that the players did not ***release*** their pushbuttons simultaneously.

Lights 5 and 6 ***both*** on indicate:

That Team B was first, but the players were not coordinated. Or, after Team B has won, that the players did not ***release*** their pushbuttons simultaneously.

The program for a team reaction tester is:

1A/3F
1A/3L
1B/2B
1E/3Z
1E/2E
1F/2A
1F/1G
1G/5J
1H/1+
1H/2H
1K/2K
1L/1+
1L/2L
1X/2X
1Y/2Y
1Z/2Z

2B/3B
2E/3C
2F/2G
2F/5A
2G/5N
2H/3H
2K/3K
2X/4J
2Y/5H
2Z/5J
2cap/3L
3A/4A
3A/6K
3B/4B
3C/4C
3F/3G

$\begin{pmatrix} 3G/\Omega_{22} \\ \Omega_{22}/3cap \end{pmatrix}$
3H/4H
3J/6X
3K/4K
3N/5G
3Y/3−
4B/5B
4C/5F
4F/4G
4F/6A
$\begin{pmatrix} 4G/\Omega_{22} \\ \Omega_{22}/4cap \end{pmatrix}$
4K/D19
4L/6A
4L/5cap

4N/5K
5B/6B
5E/6G
5F/6F
5L/6Y
5N/6Z
5X/6X
5Y/6Y
5Z/6Z
6B/6F
6E/6G
6E/D1
6H/6+
6L/6+
6L/D18
D16/D18
D17/M−

To play:

Each team should be ready at its pushbuttons. Turn power on. Push the re-set button (push-button 3) and hold it down for a full revolution of the rotary switch. As soon as lights 3 and 4 come on, each team tries to be first to push its pushbutton down simultaneously.

If any player pushes his pushbutton **before** lights 3 and 4 come on, MINIVAC will indicate a foul for that person's team and the game must be started again.

The results of the game will be displayed on the output lights.

To start another game, push the re-set button and hold it down for a full revolution of the rotary switch.

Note: To prevent anticipation of the start signal, either make dummy connections to the rotary switch or cover the dial with an inverted box.

Experiment 23: Black Jack

This program allows MINIVAC to play the card game "Black Jack" against human opponents. The players must, of course, do all the shuffling and dealing. One player will have the advantage of knowing MINIVAC's hand, since he must "read" the hand into the computer. (The game "Black Jack" is also known as "Twenty One" or "Vingt et Un.")

The Program for Black Jack is:

1A/1K	2J/2+	4E/4−	5F/6K
1B/1−	2L/5A	4E/4G	(5H/Ω_{22} Ω_{22}/6N)
1B/1E	2L/5H	(4F/Ω_{B4} Ω_{B4}/4J)	5cap/6L
1F/1G	2cap/3L	4F/5N	6A/6H
1F/1L	3C/3+	(4H/Ω_{22} Ω_{22}/5K)	6B/6−
(1G/Ω_{22} Ω_{22}/1cap)	3E/3−	4J/5+	6B/6E
1H/1+	3E/3G	4cap/5L	6C/6+
1N/2K	(3F/Ω_{B3} Ω_{B3}/3J)	5B/5−	6E/6G
2C/2+	3F/4N	5C/5−	(6F/Ω_{B6} Ω_{B6}/6J)
2E/2−	(3H/Ω_{22} Ω_{22}/4K)	5C/5J	6F/6Z
2E/2G	3J/3+	5E/5+	(6H/Ω_{22} Ω_{22}/6X)
(2F/Ω_{B2} Ω_{B2}/2J)	3cap/4L	5E/5G	6J/6+
2F/3N	4A/4+	(5F/Ω_{B5} Ω_{B5}/5J)	6Y/6cap
(2H/Ω_{22} Ω_{22}/3K)	4B/4H		
	4C/4+		

To play:

Deal a regular Black Jack hand (one card face down, one card face up) to each player, including MINIVAC.

At MINIVAC's turn to play, "read" the values of MINIVAC's cards into the computer by pushing pushbutton 6. (Push once for an ace, twice for a deuce, etc.; all face cards have a value of ten.) Note: An ace may also have the value eleven if the player so desires. The value of an ace must be decided by the person "reading" for the computer.

If light 1 remains off, MINIVAC is indicating that it will take another card. When this card is dealt, it must be read in on pushbutton 6.

If light 1 goes on, MINIVAC is indicating that it will stand on the cards it has.

If lights 1, 4, 5 and 6 ***all*** come on at once, MINIVAC is indicating that its hand exceeds twenty-one.

To re-set the Black Jack game, turn power off, then on again.

Experiment 24: Script Writer for Westerns

Full-scale computers have been programmed to "write" scripts for western movies. The computer is given information in the form of possible characters, locations and actions; the computer may also be given specific rules so that the "good guys" will always win. Since a computer has no inherent knowledge of what is and is not done in a western movie, the scripts which it produces contain a certain number of impossible — although frequently humorous — situations.

The program which follows will allow MINIVAC to write an action script for a western. The length of the computer's tale will be determined only by your whim; given enough time, MINIVAC can prepare an action script for a western epic.

We must, of course, provide MINIVAC with sufficient information to write a script. This information will be coded:

a list of possible characters — six "bad guys" and six "good guys"
a list of sixteen possible actions
a list of sixteen possible locations.
a program which will allow the computer to write with this information.

The cast of characters is:

Good Guys (and Gals)

Code for lights 1, 2 and 3	Characters
001	Claude — the trusted Sheriff of Booletown
010	Everett — the handsome young deputy
011	Becky — the evil rancher's lovely daughter
100	Richard — the editor of the "Booletown Evening Gazette"
101	Carol — the pretty owner of the Last Chance Saloon
110	Gerard — the dynamic doctor of Booletown

Bad Guys

Code for lights 4, 5 and 6	Characters
001	Arnold — the evil owner of the A Bar A Ranch
010	Willard — the wicked bartender at the Last Chance Saloon
011	Jack — the shifty owner of the General Store
100	Bob — the crooked lawyer
101	Warren — the desperate outlaw
110	Bill — the crafty telegraph agent

The possible actions and locations are:

Code for Lights 3, 4, 5 and 6	Actions	Locations
0000	meets	Stagecoach Depot
0001	threatens	Out on the range
0010	shoots	Last Chance Saloon
0011	chases	Telegraph Office
0100	arrests	A Bar A Ranch
0101	challenges	Gazette Office
0110	jails	General Store
0111	holds up	Main Street
1000	hits	Sheriff's Office
1001	cheats	On the trail
1010	warns	Lawyer's private office
1011	proposes to	Railroad station
1100	defeats	Side Street
1101	rescues	Doctor's Office
1110	ambushes	Outlaw's Den
1111	fights	Flip-Flop Gulch

The script writer program:

1A/1K
1B/2B
1E/1+
$\begin{pmatrix} 1F/\Omega_{B1} \\ \Omega_{B1}/1- \end{pmatrix}$
1F/2H
1G/2J
1H/2F
1J/2G
1L/2L
2A/2K
2B/3B
2E/2+
$\begin{pmatrix} 2F/\Omega_{B2} \\ \Omega_{B2}/2- \end{pmatrix}$
2G/3J
2J/3G
2L/3L
3A/3K
3B/4B
3E/3+
$\begin{pmatrix} 3F/\Omega_{B3} \\ \Omega_{B3}/3- \end{pmatrix}$
3F/4H
3G/4J
3H/4F
3J/4G
3L/4L
4A/4K
4B/5B
4E/4+
$\begin{pmatrix} 4F/\Omega_{B4} \\ \Omega_{B4}/4- \end{pmatrix}$
4G/5J
4J/5G
4L/5L
5A/5K
5B/6B
5E/5+
$\begin{pmatrix} 5F/\Omega_{B5} \\ \Omega_{B5}/5- \end{pmatrix}$
5F/6H
5G/6J
5H/6F
5J/6G
5L/6L
6A/6K
6B/6−
6E/6+
$\begin{pmatrix} 6F/\Omega_{B6} \\ \Omega_{B6}/6- \end{pmatrix}$
6G/6T
6J/6W
6L/6+
6S/M+
6V/M−

Slide switch 6 RIGHT is the "select" position.
Slide switch 6 LEFT is the "display" position.

For convenience in recording the script, make a chart like this:

1 and 2 different good — bad	1 and 2 same bad — good	A	Action	B	Location

Procedure:

1. Select action and order of characters:

   Set slide switch 6 RIGHT. Give the computer enough time to make its selection; then set slide switch 6 LEFT. Read the output lights as follows:

   If lights 1 and 2 are the same (either both on or both off), place a check in the "Bad-Good" column.

   If lights 1 and 2 are different (one on and one off), place a check in the "Good-Bad" column.

   Read output lights 3 through 6 in binary code. The number displayed represents an "action" — see the coded list of actions above. Record the action which corresponds to the light display in the "Actions" column.

2. Select characters:

   Set slide switch 6 RIGHT. Give the computer enough time to make its selection; then set slide switch 6 LEFT. Read the output lights as follows:

   Lights 1, 2 and 3 display the code for the "Good Guys".

   Lights 4, 5 and 6 display the code for the "Bad Guys". (See cast of characters above.)

   Note: If there is no character corresponding to the code displayed, the machine has made an error. Set slide switch 6 RIGHT, then LEFT again.

   Record output as follows:

   If there is a check in the "Good-Bad" column, record the "good guy"in column A and the "bad guy" in column B.

   If there is a check in the "Bad-Good" column, record the "bad guy" in column A and the "good guy" in column B.

3. Select location:

   Set slide switch 6 RIGHT. Give the computer time to make its selection; then set slide switch 6 LEFT.

   Read output lights 3 through 6 in binary code. Enter the location corresponding to the code displayed in the "Location" column.

4. Repeat steps 1 through 3 for a full script.

The following is a short example of a MINIVAC script:

Claude, the trusted sheriff of Booletown, threatens Arnold, the evil owner of the A Bar A Ranch, out on the trail.

Warren, the desperate outlaw, rescues Becky, the evil rancher's lovely daughter, in the Gazette Office.

Becky chases Warren out on the range.

Gerard, the dynamic doctor, meets Bill, the crafty telegraph agent, at Flip-Flop Gulch.

Jack, the shifty owner of the General Store, threatens Everett, the handsome young deputy, at the Sheriff's Office.

Everett jails Warren at the Doctor's Office.

MINIVAC 601/6010 AND THE MINIVAC MANUAL

ARE PRODUCTS OF:

372 Main Street, Watertown, Massachusetts